The Dissertation Committee for Chia-Chih Chen
certifies that this is the approved version of the following dissertation:

Recognizing Human Activities from Low-Resolution Videos

Committee:

J. K. Aggarwal, Supervisor

Ross Baldick

Alan C. Bovik

Wilson S. Geisler

Kristen Grauman

Recognizing Human Activities from Low-Resolution Videos

by

Chia-Chih Chen, B.S.E.; M.S.E.;

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2011

Dedicated to my family.

Acknowledgments

I would like to express my sincerest appreciation to my advisor, Prof. J. K. Aggarwal. He inspires my passion for computer vision research, encourages me when I am frustrated, and trusts me in all situations. He granted me the opportunity to be part of the VIRAT project, which opened my eyes to the state of the art vision research and introduced me to many outstanding researchers.

I want to thank Prof. Grauman for being a role model to us in research. The courses I have taken from her laid an important foundation of my research background. In addition, I am grateful to Prof. Baldick, who offered me my first job at UT. He is always very supportive and generous to me. I appreciate Prof. Bovik and Prof. Geisler for their wonderful courses and valuable comments and feedback on my research.

Last but not least, I would like to express my heartfelt gratitude to my fellow lab mates including Elden Yu, Michael Ryoo, Medha Bhargava, Jong Taek Lee, Lu Xia, and Saurajit Mukherjee, with whom I have collaborated on projects and papers; and also, Josh Harguess, Birgi Tamersoy, Changbo Hu, Matthew Riley, Goo Jun, Yun Koo Chung, Chunmei Liu, and Suyog Jain. I thank them all for their discussion, support, and friendship!

Recognizing Human Activities from Low-Resolution Videos

Chia-Chih Chen, Ph.D.
The University of Texas at Austin, 2011

Supervisor: J. K. Aggarwal

Human activity recognition is one of the intensively studied areas in computer vision. Most existing works do not assume video resolution to be a problem due to general applications of interests. However, with continuous concerns about global security and emerging needs for intelligent video analysis tools, activity recognition from low-resolution and low-quality videos has become a crucial topic for further research. In this dissertation, We present a series of approaches which are developed specifically to address the related issues regarding low-level image preprocessing, single person activity recognition, and human-vehicle interaction reasoning from low-resolution surveillance videos.

Human cast shadows are one of the major issues which adversely effect the performance of an activity recognition system. This is because human shadow direction varies depending on the time of the day and the date of the year. To better resolve this problem, we propose a shadow removal technique

which effectively eliminates a human shadow cast from a light source of unknown direction. A multi-cue shadow descriptor is employed to characterize the distinctive properties of shadows. Our approach detects, segments, and then removes shadows.

We propose two different methods to recognize single person actions and activities from low-resolution surveillance videos. The first approach adopts a joint feature histogram based representation, which is the concatenation of subspace projected gradient and optical flow features in time. However, in this problem, the use of low-resolution, coarse, pixel-level features alone limits the recognition accuracy. Therefore, in the second work, we contributed a novel mid-level descriptor, which converts an activity sequence into simultaneous temporal signals at body parts. With our representation, activities are recognized through both the local video content and the short-time spectral properties of body parts' movements. We draw the analogies between activity and speech recognition and show that our speech-like representation and recognition scheme improves recognition performance in several low-resolution datasets.

To complete the research on this subject, we also tackle the challenging problem of recognizing human-vehicle interactions from low-resolution aerial videos. We present a temporal logic based approach which does not require training from event examples. At the low-level, we employ dynamic programming to perform fast model fitting between the tracked vehicle and the rendered 3-D vehicle models. At the semantic-level, given the localized event

region of interest (ROI), we verify the time series of human-vehicle spatial relationships with the pre-specified event definitions in a piecewise fashion. Our framework can be generalized to recognize any type of human-vehicle interaction from aerial videos.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Owing to the increasingly ubiquitous presence of video cameras, a gigantic amount of video data is being generated daily and is awaiting further analysis for different purposes. Humans are the center of interest in most videos, and vision based human activity recognition has found many applications such as active surveillance, video indexing, and human computer interactions. An ideal human activity recognition system is expected to accurately localize, segment, and semantically annotate continuous activities of multiple agents in unconstrained environments. However, even with state of the art computer vision algorithms, the accurate recognition of human activities from real-world videos remains an ambitious task.

Here we enumerate several major challenges to the recognition task. First is intra-class variability and inter-class similarity of activities. Individuals can perform an activity in different directions with different characteristics of body part movements, and two activities may be only distinguished by very subtle spatio-temporal details. Second, the number of describable activity categories is huge; the same activity may have different interpretations under different object and scene contexts. Third, occlusions, background, cast

(a) (b) (c)

Figure 1.1: Representative applications of low-resolution human activity recognition: (a) aerial video analysis, (b) wide area surveillance, and (c) sports video analysis.

shadow, illumination condition, and view point changes can all alter the way activities are perceived. Therefore, to successfully recognize activities under the given assumptions, the system should have robust low-level preprocessing techniques, an activity descriptor which provides both invariance and descriptive power, and the appropriate mathematical model which takes contextual information into the decision making behind classification.

The scope of my research in this thesis is the recognition of human activities from low-resolution videos. By low-resolution, I mean "low human figure resolution", which in general ranges from 20 to 40 pixels in height. The subject human activities that we aim to recognize include: (1) single person actions, (2) composite human activities, and (3) human-vehicle interactions. There are several applications that require the analysis of human activity from low-resolution imagery, for example, aerial video analysis, wide area surveillance, and sports video annotation and searches, as shown in Figure 1.1.

Low-resolution activity recognition features other challenges in addition

Figure 1.2: A (a) 140-pixel tall human figure is downsampled to (b) 50-pixel, (c) 20-pixel, and (d) 10-pixel high figures. We aim at recognizing human activities from imagery of which the resolution is between (b) and (d).

to those presented in general purpose activity recognition tasks. Figure 1.2 [58] for example, illustrates how a person appears in different resolution settings. This figure shows that when a human figure is lower than 20 pixels in height, the configuration of the person's limbs is barely recognizable. Even the object categorical information will be missing when the image resolution is as low as in Figure 1.2(d). Our methodologies are *not* developed to recognize human activities from super low-resolution imagery (say Figure 1.2(d)), where the trajectory level analysis of activities may be the only viable solution.

Unfortunately, limited visual information extracted from low-resolution imagery alone does *not* account for all the causes of significant accuracy reduction in low-resolution activity recognition. We summarize 3 other major difficulties in this task. First, low-resolution videos are usually filmed outdoors at a distance; therefore, the local edge structure or appearance of human fig-

ures tend to be vague due to the blurring effect of air turbulence. Second, the orientation of a human cast shadow varies as a function of time, which changes the extracted visual patterns from the video sequences of the same activity. Third, the performance of an activity recognition algorithm depends largely on the quality of the tracks of human and other objects, which is far less reliable in low-resolution imagery as compared in regular resolution videos. For example, low-resolution aerial videos are filmed by moving cameras, which may frequently change view points, zoom levels, and field of view (FOV). As a result, video stabilization is a common preprocess, a computed track can be broken into pieces, and an activity sequence contains images from different views of the tracked person.

There has been abundant literature on vision based activity recognition. Some of the methodologies rely on the exact characterization of human contours [9, 40], extremities [31, 87], or parts (limbs, head, torso, etc.) [11, 63], while others assume low-level preprocessing not being a critical issue that adversely effects recognition accuracy. However, from our experimental results, the direct application of the existing techniques to the low-resolution datasets [15, 28, 58] usually ends up with near random performance. In view of the aforementioned challenges, we have proposed a methodology to remove human cast shadows in any direction, a joint feature histogram based action descriptor, a novel mid-level feature and activity recognition scheme, and a 3-D vehicle model fitting technique together with a temporal logic based approach for human-vehicle interaction reasoning.

1.1 Preview of the Chapters

There are generally 4 computational steps [1] to recognize real-world human activities from videos, which include human detection, human tracking, human action recognition, and then higher-level activity evaluation. Each component plays an essential role both for the subsequent processing unit and for the overall system performance. The focus of my contribution is on the processes *after* acquiring the human and object (vehicle in this work) tracks. On top of the proposed activity representations and recognition algorithms, we present our low-level preprocessing techniques which predominate the recognition accuracy in the low-resolution video setting.

1.1.1 Preprocessing

The goal of our preprocessing steps is to refine the tracking results so that the input to our activity recognition modules is shadow-free figure centric bounding box sequences. To compute figure-centric bounding boxes [15], we perform human detection in the local neighborhood of the track coordinates using Histogram of Oriented Gradients (HOG) [23] as feature and linear Support Vector Machines (SVM) for human detector. For each frame of a human track, we extract activity features from the bounding box of which the scale and translation corresponds to the highest probability estimate.

Furthermore, we present a shadow removal technique [16] which effectively eliminates a human shadow cast from a light source of unknown direction. A multi-cue shadow descriptor is proposed to characterize the distinctive

properties of shadows. We employ a 3-stage process to detect and then remove shadows. Our algorithm improves the shadow detection accuracy by imposing the spatial constraint between the foreground subregions of human and shadow. We collect a dataset containing 81 human-shadow images for evaluation. Both descriptor receiver operating characteristic (ROC) curves and qualitative results demonstrate the superior performance of our method.

1.1.2 Human Action Recognition by Combining Discriminative Low-level Features

In this chapter, we introduce a new descriptor [15] to characterize human actions when they are being observed from a far field of view. In our representation, an action sequence is divided into overlapped spatial-temporal volumes to make robust and comprehensive use of the available features. Within each volume, we represent successive poses by time series of HOG and movements by time series of Histogram of oriented Optical Flow (HOF) [24]. Supervised Principle Component Analysis (SPCA) [68] is applied to seek a subset of discriminatively informative principle components (PCs) to reduce the dimension of histogram vectors without loss of accuracy. The final action descriptor is formed by concatenating sequences of SPCA projected HOG and HOF features. A Support Vector Machines (SVM) classifier is trained to perform action classification. We evaluated our algorithm by testing it on 2 regular resolution and 3 low-resolution datasets, and compared our results with those of other reported methods.

1.1.3 Human Activity Recognition with Mid-Level Features and Speech-Like Processing

In the previous chapter, the proposed action descriptor is built at the scale of the entire human figure based on a combination of shape and motion features; however, there is room for improvement to be made. For example, recognizing activities from sparse and noisy low-level features alone limits the performance. Therefore, we developed a novel mid-level feature [17], action spectrogram (AS), which is built from local gradient and optical flow features learned in a discriminative manner. Our representation is inspired by a common spectrographic representation of speech. Different from sound spectrogram, an action spectrogram is a space-time-frequency representation of actions, which enable us to model the spectral properties of body parts' movements in addition to verifying features from local video content.

Our method first converts an activity sequence into multiple likelihood time series of action associated local interest patterns. These simultaneous temporal signals from active body parts are divided into overlapped short time segments and converted by an 1D Fast Fourier Transformation (FFT) to synthesize a volume of AS. Then slice by slice we classify an AS volume into sequential actions, and model their temporal evolution via activity Hidden Markov Models (HMM). We have tested our algorithm on a variety of human activity datasets and achieved superior results.

1.1.4 Human-Vehicle Interaction Recognition Without Event-Level Training

In this chapter, we detail our framework for human-vehicle interaction recognition [49] from low-resolution aerial videos. In this scenario, the object resolution is low, the visual cues are vague, and the detection and tracking of objects are less reliable as a consequence. Any methods that rely on the accurate tracking of persons or the exact matching of event definitions are best avoided. To address these issues, we propose a bottom-up approach which does not require the tracking of human objects nor the training from event-level examples. At the low-level, we localize a sequence event Region of Interest (ROI, vehicle doors in this work) via 3-D vehicle model alignment under a dynamic programming formulation. This is achieved by training vehicle location and orientation classifiers with synthesized 3-D vehicle images. The classification results are intergraded into our dynamic programming scheme to compute the optimal vehicle alignment parameters over the entire vehicle sequence. At the higher-level, we propose a modified temporal logic algorithm to reason about the human-vehicle interaction which happened among the localized ROI sequence. The spatio-temporal relationships between the detected person and the event ROI are verified with the manually encoded interaction definitions in a piecewise fashion. We demonstrated the performance of our method on a subset of the VIRAT Aerial Video dataset [58] for the interactions of a person getting into and getting out of a vehicle. Our approach is shown to be rather robust and can be easily extended to recognize any type of human-vehicle

interaction.

1.2 Main Contributions

My contributions to low-resolution human activity recognition can be presented in 3 levels from a system perspective. For low-level image processing, I propose to refine a noisy human bounding box sequence with the preprocesses of human figure centralization and human cast shadow removal. I develop a multi-cue shadow descriptor to characterize the distinguishing properties of shadow pixels invariant of the light source. The proposed 3-stage process segments a ray human shadow as an intact region based on pixel-level detection, which largely reduces the risk of classifying human pixels as shadow. Our experimental results show that the proposed log-polar coordinate and single-cell HOG features outperform the HSI (hue, saturation, intensity) color feature in the measure of area under the ROC curves (AUC).

Given a refined sequence of human bounding boxes, I present two approaches to recognize single person activities. The descriptor in the first approach encodes human poses and motion feature time series in a discriminative way to provide a comprehensive yet efficient representation of low-resolution action sequences. I extend supervised PCA [68] to select features in a multi-class problem. To further the performance on low-resolution imagery, I present a novel mid-level feature which characterizes activities with both local video content and likelihood spectra of body parts' movements. I propose the idea of learning action associated local visual patterns from detected spatio-temporal

interest points (STIP). The boosted weak learners are mostly localized on action associated body parts. My speech-like modeling of human activities facilitates the evaluation activities using linguistic-like models. Experimental results show that with the additional cue of body parts' spectral properties, the recognition accuracy is improved 2 to 4 percent (see Section 5.3) on the tested low-resolution datasets.

For high-level human-vehicle interaction analysis[1], we have made two major contributions to accommodate the issues from the low-resolution aerial view setting. First, we propose a dynamic programming based 3-D vehicle model alignment technique, which searches for the optimal vehicle translation and orientation parameters over the entire sequence. Comparing to common template matching based approaches, to avoid the extraction of unreliable edge maps or backgrounds, we train two separated gradient feature classifiers for vehicle state estimation. Second, at the semantic level, we propose the algorithm named piecewise temporal logic (PTL) to derive interaction sub-events from event states. PTL enables us to bypass the direct recognition of interaction sub-events from video sequences, which is highly inaccurate in our scenario due to low figure resolution, salient vehicle edge structures, and time varying view points. Furthermore, the use of a temporal logic based approach saves the cost of manually collecting and labeling the training examples from aerial videos.

[1]This is a joint work with Mr. Jong Taek Lee. My contribution to this work is on overall methodology scheming and interaction analysis from event ROI.

Chapter 2

Related Work

There has been a significant amount of research on human activity recognition with a variety of feature representations and recognition schemes under different scenario setups. The goal of this thesis is to provide a systematic way to introduce our novel activity descriptors and interaction representation scheme which are developed to address the challenges posed by low-resolution imagery. I organize this chapter into 4 sections, which in turn cover the literature regarding (1) related work to our human shadow detection algorithm, (2) previous work on low-resolution activity recognition, (3) related work to our mid-level speech-like activity representation, and (4) existing work on human-vehicle interaction recognition.

2.1 Human Shadow Detection

The paper by Prati *et al.* [60] provides a comparative survey on the literature of shadow detection. In their survey, color is the most commonly used feature for shadow characterization. When comparing a shadow pixel with its unshaded neighborhood, the most obvious observation is the difference in luminance. Shadows not only reduce the luminance of the shaded area

but also distort its chromaticity. To lower the dependence of chromaticity on luminance, several methods [30, 67, 76] have been proposed to exploit the invariant color spaces. Other than the manipulation of color space, by a rule based method, Cucchiara *et al.* [21] detect shadow by thresholding the HSV distance between the image and the background model. They apply different distance metrics for individual color components. However, this work assumes the availability of background which needs be updated or retrained when the time of the day or the scene is changed.

Zhu *et al.* [90] tackles a more challenging problem on recognizing shadows from monochromatic natural images. Their motivation is that color may not be available in all types of sensors, and they aim to understand the extent of how monochromatic cues can be explored. They propose to extract 3 groups of features from image segments, which include shadow variant features, shadow invariant features, and near black features. These features (8 in total) are used to train a classifier from boosting a decision tree and integrated into a Conditional Random Field (CRF), which improves shadow segmentation results by enforcing local consistency over pixel labels. However, there is no analytical results about the contribution of each feature and how segmentation parameters may effect the performance.

More on detecting shadows from image segments, Guo *et al.* [36] propose to learn pairwise illumination relationships between segmented regions and apply graphic cut to acquire the labeling of shadow and non-shadow regions. The features they extract from a single image region include a L*a*b

color histogram and a texton texture histogram [55]. Based on that, they classify image region-region relationships into same illumination pairs, regions that are of the same material and illumination, and different illumination pairs, regions that are of the same material but of different illumination. The illumination conditions of the graphic-cut merged shadow regions are then recovered by image matting. Image segmentation and pairwise region relationship classification processes may limit the real-time application of the algorithm.

Besides color, other features have also been shown to be useful for shadow characterization. In [7], for example, an edge map is used to segment the image into edged, smooth, and textured regions. The smooth region is regarded as the candidate area of a shadow. Edge orientation and other shadow related geometric properties are adopted as features in [12]. The work by Rielly *et al.* [61] approaches the problem from a quite different perspective. They aim to detect humans in aerial imagery using shadow cast as an auxiliary cue. The metadata acquired from aerial vehicle is used to compute the orientation of groundplane normal and the length and orientation of human cast shadows. After a series of image processes, the binary map of blobs with strong gradient magnitude in the normal of human and shadow directions are computed. As shown in Figure 2.1(a), a candidate human blob (green) is paired with the nearest shadow blob (red) to form a candidate human-shadow blob. Figure 2.1(b) shows that the candidate human-shadow blobs are further trimmed down to a smaller set of pairs by verifying the human-shadow included angles with metadata. Humans are detected from the image patches

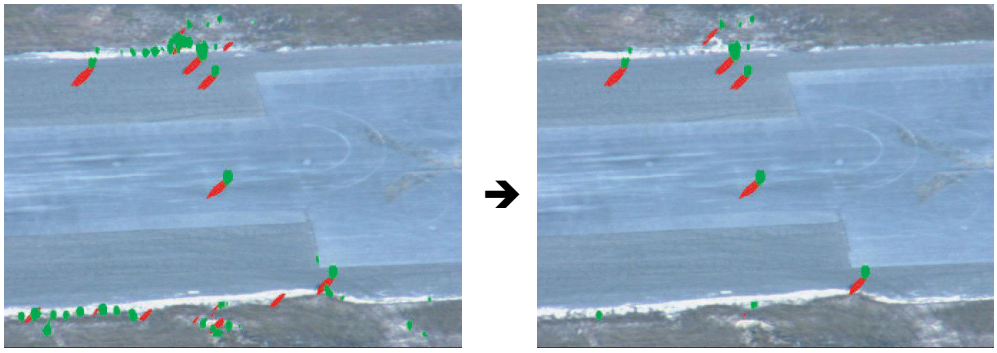


Figure 2.1: The intermediate results of [61]. Shadow information is taken into account for human detection. (a) Detected human (green) and shadow (red) blobs using image gradient and metadata. (b) The human-shadow blobs that satisfies the geometric constraints derived from metadata.

of the geometrically qualified human blobs using a combination of wavelet features and a SVM classifier. However, this method is likely to fail on a cloudy day when there are no strong gradient vectors can be extracted from human cast shadows.

2.2 Low-resolution Activity Recognition

The survey papers by Aggarwal and Cai [2], Aggarwal and Ryoo [3], Gavrila [33], and Hu *et al.* [41] provide an extensive review of representations and algorithms for human tracking, motion analysis, action representation, and activity analysis. In this section, we look specifically at the work which addresses the challenges in low-resolution activity recognition or adopts similar representation to our approach [15].

The paper by Efros *et al.* [28] is known to be the first work which devices their descriptor for representing human actions at a distance. They aim to recognize actions from sequences of human figures that are only about 30 pixels tall. The proposed motion descriptor is computed from smoothed and aggregated optical flow vectors over a spatio-temporal volume centered on a moving person. They use a k-nearest-neighbor classifier to perform action recognition and synthesis. Nevertheless, the use of motion feature alone is insufficient to characterize certain "static" actions. Moreover, they compute the optical flow feature between figure-centric frames, which implicitly removes the velocity information of human movement.

The recent work by Ahad *et al.* [4] presents a new descriptor called directional motion history image (DMHI) to characterize low-resolution poor-quality human activities. Their representation is an extension of motion history image (MHI) [9]. The major difference is that MHI computes the update function from frame differencing or background subtracted images while DMHI uses optical flow. Similar to [28], in this representation, an optical flow filed is divided into 4 different direction maps; from which 4 DMHI images are computed. They employ a k-nearest-neighbor classifier for action recognition. Despite the existence of several public low-resolution activity datasets, they tested their method only on a self collected *indoor* activity dataset with downsampled frame resolutions. This experimental setting does not provide a candid assessment of their method, because most low-resolution poor-quality videos are filmed outdoors at a distance with the blurring effect from air tur-

bulence.

We compiled and published the UT-Tower dataset [15] for the evaluation of activity recognition algorithms on a more realistic collection of low-resolution human action footage. This dataset was filmed top-down from a 307 feet high tower building to simulate the imagery taken from an aerial vehicle. Later on, testing on the UT-Tower dataset, Ryoo *et al.* [65] held the "Aerial View Activity Classification Challenge" to motivate researchers to explore techniques that better accommodate the challenges inherent in arial view imagery. There were 4 university teams [35, 77, 86] who participated in this contest. The results show that the performance of the competing algorithms are about the same as the baseline method, which is a combination of a spatio-temporal HOG descriptor and a linear SVM classifier. The winner of the contest is the method named 'action covariance manifolds' proposed by Guo et al. [35] from the Boston University team. They represent a sequence of human actions as the shape of the silhouette tunnel, which is a temporal sequence of local shape-deformations of centroid-centered object silhouettes. The empirical covariance matrix is a set of 13-dimensional feature matrix extracted from the silhouette tunnel. The silhouette tunnel of a test video is broken into short overlapping segments and classified using a dictionary of labeled action covariance matrices with the nearest neighbor rule.

Similar to our action representation [15], Ikizler *et al.* [42] use both human contours and motion features for action recognition. They characterize human contours by histograms of Hough transformed edge, and use coarse

orientation bins to compute optical flow distribution. They train separate shape and motion classifiers and combine both classification results by averaging them. However, there is no evidence that shape and motion features are equally useful for distinguishing actions. Therefore, the linear combination of single feature trained classifiers may not be the optimal way of improving a joint decision. In [54], Lu and Little employs the Principle Component Analysis (PCA) projected HOG descriptor in a hybrid HMM classifier for the joint task of tracking and action recognition from low-resolution sports videos. The space searched by PCA provides an efficient representation of the data, but it does not necessarily allow better separation of descriptor vectors from different actions.

2.3 Mid-level Speech-like Activity Representation

For the purpose of object recognition, features computed from local image patches [39, 53] have been shown to be invariant to certain types of image transformations and robust against cluttered background and object deformation. Recent years local interest features detected from 2-D image patches or 3-D video cuboids have also been applied extensively for activity representation. These local features are extracted from video frames either because they present spatially and/or temporally distinguishing visual patterns or they correspond to human body parts. To compute features explicitly from body parts, Chakraborty *et al.* [11] and Yu and Aggarwal [88] train separated body part detectors and recognize consecutive human poses by the configurations

of localized parts. Ryoo and Aggarwal [62] divide a human figure into head, upper-body, and lower-body parts, which are fitted with ellipses and convex hulls to estimate the corresponding state of partial poses.

On the other hand, features localized by 2-D [37, 53] or 3-D interest point detectors [27, 47, 83] do not necessarily capture specific body parts or body parts' motion patterns; instead, the detected video cuboids are mostly triggered by periodic motion, spatio-temporal corners, or any spatially interesting patterns that evolve in time. Various work [46, 48, 57, 64, 69] has explored the usage of local video features for activity representation. Recent work by Wang *et al.* [81] provides a comprehensive performance evaluation on different combinations of popular local spatio-temporal feature detectors and descriptors. Ke *et al.* [46] and Laptev and Pérez [48] boost a cascade of space-time window classifiers to recognize actions. To make the run time scalable, their weak learners are trained on features extracted from random cuboids of dense video grids. Neibles *et al.* [57] represent an image sequence as a bag of video words. Under their unsupervised learning framework, action recognition and localization are performed by maximizing the posterior of learned category models. Ryoo and Aggarwal [64] propose a kernel function to measure the structural similarity between the sets of spatio-temporal interest points (STIP) extracted from two videos. Their kernel is a histogram which bins the pairwise spatio-temporal relationships among the video words.

Our speech-like representation of activities [17] is considered to be a type of mid-level feature [27, 29, 57, 64], which is built upon low-level features

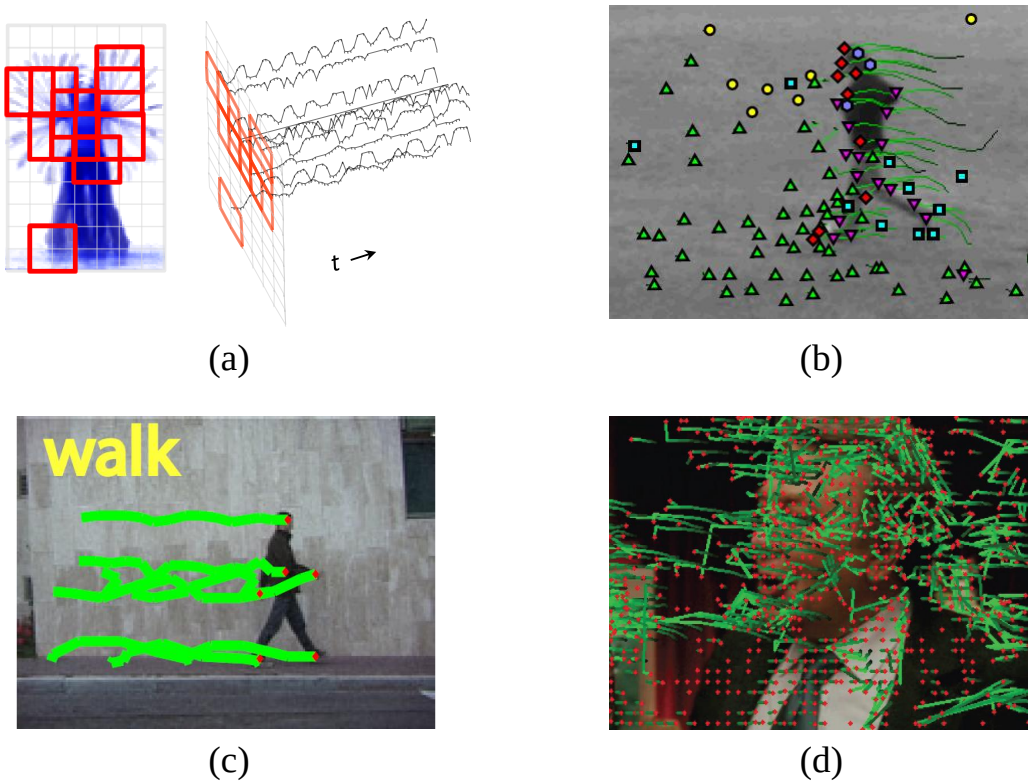


Figure 2.2: Different trajectory-like features: (a) ours, occurrence likelihood time series of local interest patterns [17], (b) tracjectons: trajectories computed from feature trackers [56], (c) trajectories of body reference joints [5], (d) trajectories of densely sampled points using optical flow fields [80].

such as image gradient, edge, or optical flow. Compared to features that describe the entire human figure [9, 28], mid-level features are focused on local regions of an action sequence to provide an efficient representation. The direct use of any detected mid-level features may lack of descriptive power in representation; therefore, Laptev and Pérez [48] and Fathi and Mori [29] have proposed different approaches for learning discriminative mid-level features.

As shown in Figure 2.2(a), our speech-like representation is computed by converting an activity sequence into multiple likelihood (of local interest patterns) time series at associated body parts. These likelihood time series appear to be very similar to the trajectories of body parts; however, they are essentially different. For example, Matikainen *et al.* [56] employ a KLT (Kanade-Lucas-Tomasi) feature tracker to track a number of features over an activity sequence (Figure 2.2(b)). The trajectories of the tracked features are processed and divided into snippets called *trajectons*. Under a bag-of-words framework, tracjectons of a video are matched against a pre-clustered trajecton library and accumulated into a histogram-based action descriptor. Different from [56], Ali *et al.* [5] assume that feature tracking is a relatively well-solved problem so that the trajectories of body reference joints can be reliably used as a feature (Figure 2.2(c)). The focus of their method is to model the nonlinear dynamics of human actions by the theory of chaotic systems. Wang *et al.* [80] propose an efficient way to extract dense trajectories (Figure 2.2(d)), which are computed by tracking densely sampled points using optical flow fields. They employ a modified motion boundary histograms (MBH) [25] for trajectory representation and a standard bag-of-feature approach for activity classification.

In previous work on spectral analysis of human action, Cutler and Davis [22] detect periodic motion by analyzing the power spectrum of the sequence self-similarity matrix. Weinland *et al.* [82] propose a free viewpoint action descriptor based on Fourier analysis of motion history volumes. The shift

invariant property of FFT enables them to extract view-invariant features from cylindrical coordinates.

2.4 Human-vehicle Interaction Recognition

Comparing to the research on general human-object interaction recognition, there has been much less work on human-vehicle interaction recognition. Recognizing human-vehicle interactions is a challenging problem not only because accessing the vehicle from different parts with different manners represents different interactions but also because of occlusions and the significant variations among the views of the same interaction. To the best of our knowledge, there is no available work that tackles the problem of human-vehicle interaction recognition from low-resolution aerial videos. Therefore, we focus on presenting work that approaches the view-dependent issue as well as the existing high-level formulations of the problem.

To be able to interfere human-vehicle interactions from untrained viewpoints, Song and Nevatia [71] propose to detect and track vehicles using 3-D vehicle models. This is achieved by extracting the planar projections of a 3-D vehicle model from various views and matching the 2-D shape templates against the candidate foreground blobs. Given the initial vehicle alignment results, a data driven Markov Chain Monte Carlo (DDMCMC) process is applied to refine the hypothesis about vehicle types, positions, and orientations. Lee *et al.* [50] present a novel approach to recognize human-vehicle interactions. Through the use of synthetic 3-D vehicle models, their system is able

to achieve view-independent recognition of human-vehicle interactions. The human-vehicle interaction is decomposed into atomic level interactions. The complete human-vehicle interaction is recognized by verifying the temporal structure of the atomic interactions. The interactions considered include entering and exiting a vehicle.

On the high-level modeling of human-vehicle interactions, Ivanov and Bobick [43] use stochastic context-free grammars to represent human-vehicle interactions. Tran and Davis [75] employ Markov logic networks to recognize human-vehicle interactions. Joo and Chellappa [45] propose to define human-vehicle interaction in terms of attribute grammars, which are capable of describing features that are not easily expressed by finite symbols. Ryoo *et al.* [66] develop a probabilistic framework to track humans and analyze their dynamic relationships with vehicles. The use of event context in their formulation enables the system to analyze states of scenes composed of multiple objects and to process complex interactions consisting of several sub-events from multiple agents.

However, all the mentioned approaches are not directly applicable to our imagery, where the interactions are filmed top-down from a moving platform and the accurate characterization of object contour and motion is not possible. In our proposed work [49], due to the difficulty of modeling video background, we train separate vehicle location and orientation detectors with synthetic vehicle examples to achieve view-independent recognition of vehicle states. At the high-level, we propose to use piecewise temporal logic to

derive interactions from noisy sub-event detection sequences. For the evaluation of human activity and human-vehicle interaction recognition algorithms, the newly published VIRAT Video Dataset [58] includes videos collected from stationary ground cameras as well as unmanned aerial vehicles (UAV). This large-scale benchmark dataset features 6 types of human-vehicle interactions in both camera settings.

Chapter 3

Preprocessing

We aim at recognizing human activities from tracks of human objects computed from low-resolution videos. The left column of Figure 3.1(a) illustrates the quality of several human tracks computed from low-resolution aerial videos using a Kalman filter based approach . The goal of the preprocessing steps is to safeguard that the input to our activity recognition modules is a spatial-temporal volume centered on the acting person without the inclusion of the person's cast shadow. This involves the computation of figure-centric bounding boxes and the removal of human shadows casted from any directions of sunlight. I detail each of the preprocessing component in the following sections.

3.1 Human Figure Centralization

Given a stabilized video with tracks of human objects, the purpose of this preprocessing stage is to acquire figure-centric action sequences from the tracks. The figure-centric representation implicitly aligns human body parts over a spatial-temporal volume so that activities are recognized by the relative changes in parts' configurations instead of simple translations. This step is

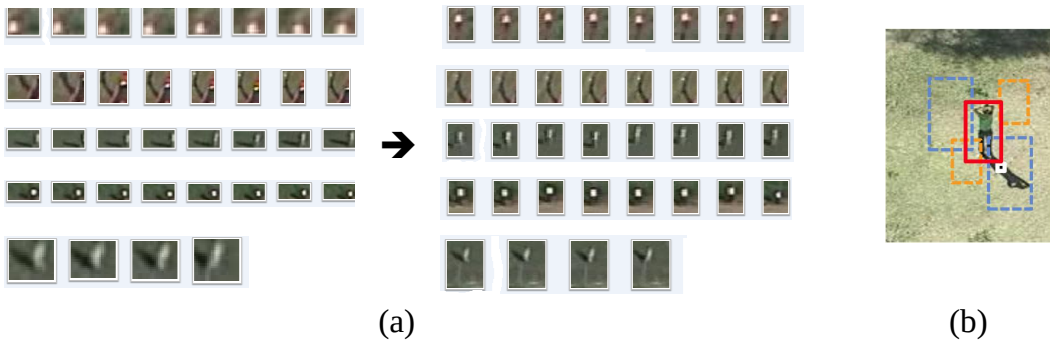


Figure 3.1: Human figure centralization. (a) Tracks of human objects before (left) and after (right) the figure centralization process. (b) Given the track coordinate (white square) the bounding box for HOG extraction (red) is centered on the human figure by searching in the space of scale and translation.

critical, because in low-resolution video frames, even a minor misalignment of a bounding box can cause the loss of body parts or a large inclusion of background. To overcome this difficulty, we take the approach similar to [23] for human figure centralization. The major difference is that, instead of searching for all people in the entire frame, it is assumed that the person of interest is somewhere around the track coordinate. We train our figure centralization detector with HOG descriptors extracted from manually cropped figure-centric bounding boxes and negative samples from descriptors of patches around the figures. During runtime, within the neighborhood of interest, the detection window searches in the space of scale and translation (Figure 3.1(b)). For a specific scale and translation which the SVM window classifier computes the highest probability estimate, the corresponding HOG vector and the window coordinates are stored. The localized bounding box coordinates are then used for the computation of HOF. As shown in right column of Figure 3.1(a), my

approach improves the quality of bounding boxes coverage of most computed tracks.

3.2 Human Shadow Removal with Unknown Light Source

The existence of human shadows is a general problem in tracking and recognizing human activities. Shadows not only distort the color properties of the area being shaded but also complicate the edge structure of the figure as a whole. There are several factors that together determine the appearance of a shadow, for example, the view point of camera, the angle of incidence, the light intensity, and the number of light sources, etc. Further, under the sun, the dominant orientation of a human shadow changes as a function of time. Therefore, a human tracker becomes more prone to miss the target, and the motion pattern of a single action varies considerably. For simplification, by human shadow we mean a human cast shadow in contrast with a human self shadow.

The purpose of this work is to replace the region of a human shadow with the estimation of underlying unshaded background. Without loss of generality, we assume that human figures are posed vertically and the foreground mask is available to us. We simplify this problem by taking advantage of the fact that both human and shadow regions within a foreground blob are connected components. The task of shadow detection can thus be posed as a search for the linear boundary which best separates the two connected subregions. We propose a bottom-up classification scheme to approximate the

optimal boundary. The preliminary classification is to divide foreground pixels into the intermediate classes of *shadow* and *non-shadow*. Based on the pixel locations of the labeled pixels, the secondary classification segments a connected shadow region from the foreground blob. Finally, we inpaint the detected shadow region with a Gaussian spatial filter. For reliable characterization of a shadow pixel, we extract three types of features from each pixel and its neighborhood. These features include color, relative pixel location, and local HOG.

This section is organized as follows: subsection 3.2.1 introduces our shadow descriptor. The proposed process for shadow removal is presented in subsection 3.2.2. We demonstrate our experimental results in subsection 3.2.3 and conclude in subsection 3.2.4.

3.2.1 Characterization of Shadow Pixels

Most existing work on shadow detection [30, 67, 76] uses color as the major or the only cue to characterize shadows. However, in real-world imagery, shadows have a wide spectrum of luminance values. Therefore, shadow detectors that mainly rely on color information are more susceptible to the changes in lighting conditions. In this work, we use a multi-cue descriptor to represent human shadows. The proposed descriptor is the concatenation of three normalized shadow distinctive features, which are detailed as follows:

Color. For accurate detection of shadows, the choice of color space is important. Various color spaces have been explored to search for a transforma-

tion which provides better discrimination of projected pixels or least effects of shadows on chromaticity. Both HSI and HSV (V for value) are popular color models in literature. Particularly, in Tsai's [76] experiments on 6 color spaces, the highest detection rate is achieved by remapping color into HSI space. Following his results, we transform RGB color into HSI space by

$$\begin{bmatrix} I \\ V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{-\sqrt{6}}{6} & \frac{-\sqrt{6}}{6} & \frac{\sqrt{6}}{3} \\ \frac{1}{\sqrt{6}} & \frac{-2}{\sqrt{6}} & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.1}$$

$$S = \sqrt{V_1^2 + V_2^2} \tag{3.2}$$

$$H = \tan^{-1}(V_2/V_1), V_1 \neq 0 \tag{3.3}$$

Log-polar coordinates. Connecting to the bottom of human figures, shadows appear in various orientations and shapes. We find that pixel locations in Cartesian coordinates are less informative about the coverage of a shadow. Therefore, we devise a modified log-polar coordinate system to make better use of pixel location as a feature.

Motivated by the non-uniform mapping from a human retina to visual cortex [70], a log-polar coordinate system is preferable to a Cartesian system in certain applications. In log-polar coordinate system, the origin area has a higher resolution as compared to the periphery. We modify the system in a way that the distribution of coordinate resolution approximates the confidence map of shadow coverage. As shown in Figure 3.2, the modified log-polar coordinates are superimposed onto a human-shadow foreground blob. Via this representation, shadow pixels will occupy mostly the low-resolution peripheral

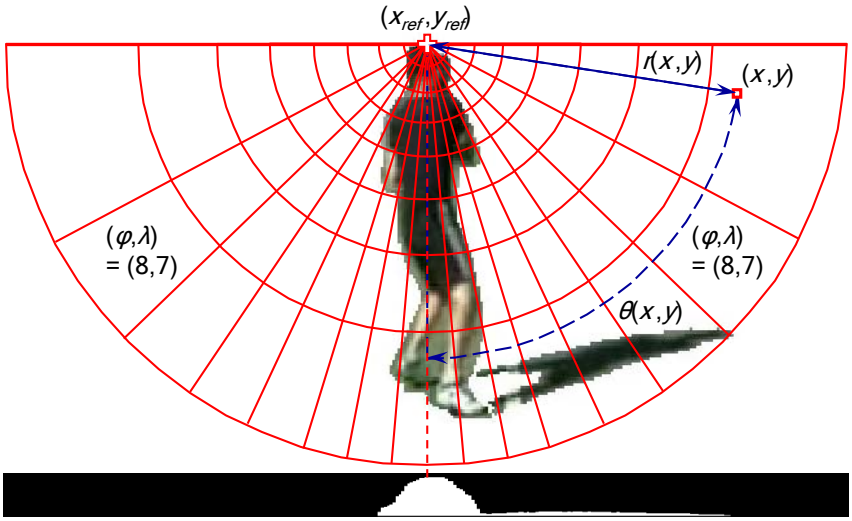


Figure 3.2: Top: diagram of the modified log-polar coordinate system. Bottom: horizontal projection histogram.

area, where the certainty about a shadow's presence is low.

A reference point, (x_{ref}, y_{ref}), is first located before the computation of the projected pixel location. Here y_{ref} is equal to the y-coordinate of the top of the foreground blob. x_{ref} corresponds to the x-intercept of the major axis of a person's silhouette. We compute x_{ref} by locating the peak of the horizontal projection histogram, which is an accumulation of foreground pixels along the vertical axis (Figure 3.2). The radius, R, is the maximum distance from the reference point to the blob boundary. Θ is a fixed value which is set to $\pi/2$. The radial and angular resolution of the coordinate system are symbolized by

Φ and Λ, respectively. The mapping from (x,y) to (ϕ,λ) is defined as

$$\phi(x,y) = \left\lceil \Phi + 1 - (\Phi+1)^{1-r(x,y)/R} \right\rceil \tag{3.4}$$

$$\lambda(x,y) = \left\lceil \Lambda + 1 - (\Lambda+1)^{1-\theta(x,y)/\Theta} \right\rceil \tag{3.5}$$

where

$$r(x,y) = \sqrt{(x-x_{ref})^2 + (y-y_{ref})^2} \tag{3.6}$$

$$\theta(x,y) = \tan^{-1} |(y-y_{ref})/(x-x_{ref})| \tag{3.7}$$

HOG. We use orientation transformed single-cell HOG as one component feature for two reasons. First, the dominant edge orientation of a human local silhouette is mostly close to vertical, while the dominant orientation of a shadow can be in all directions. Second, strong edge structure is not always available from the region of a shadow [18].

As shown in Figure 3.3, a human figure is connected with shadows oriented in various directions. The arrows and square areas represent gradient vectors and HOG cells, respectively. For the cells on the human figure (human HOG), the corresponding HOG vectors are expected to have greater values over the horizontal bins. However, the maximum bin of a shadow region HOG vector (shadow HOG) is closely related to the dominant orientation of the shadow.

In [23], unsigned gradient vectors are used for HOG computation. However, there is one problem with the original HOG representation, which adversely effects detection accuracy. In a human HOG, there are two bins (0

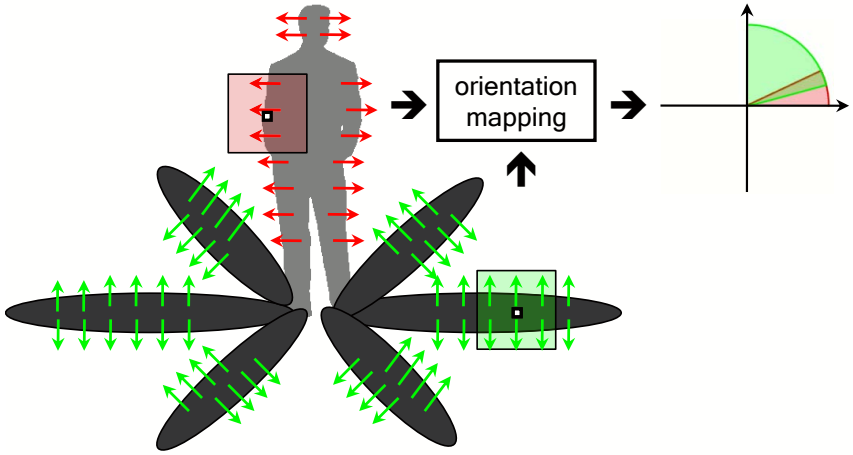


Figure 3.3: Left: Single-cell HOG on human and shadow subregions. Right: Schematic drawing of the Eq. 3.8 projected human (red) and shadow (green) gradient vectors on a polar coordinate.

and π) that correspond to the horizontal direction. Therefore, a horizontal gradient vector may vote for either of the two bins depending on the sign of human-background intensity difference and the cell location (left or right body parts specifically). To solve this problem, we apply the following mapping.

$$\phi = \frac{\pi}{2} - \left| \frac{\pi}{2} - \theta \right| \tag{3.8}$$

Here θ represents the unsigned angle of a gradient vector, which is transformed into ϕ for HOG computation.

3.2.2 Shadow Detection and Removal

In this work, we aim at finding a boundary which divides a human-shadow foreground blob into its ground truth subregions. We propose a 3-stage

process to implement this idea. The first stage performs a binary classification on pixels of a foreground blob. A radial basis function (RBF) kernel SVM classifier is trained with descriptors from the labeled shadow images.

Using the first stage classification results as intermediate ground truth, the second stage computes the linear boundary within the foreground blob that minimizes the classification error. For this purpose, pixel coordinates alone are used as a feature. We adopt a linear classifier to avoid the overfitting problem from a complex decision boundary. In other words, the linear classifier divides a foreground blob into human and shadow subregions by referring to stage one labeled pixel locations. In the third stage, we inpaint the detected shadow region with the estimation of an unshaded background. A 2D spatial filter is applied to replace the color of each detected pixel with the Gaussian-weighted average of neighboring background pixel values.

As can be imagined, both feature extraction and nonlinear classification are time consuming processes. Therefore, without loss of accuracy, we use a downsampled human bounding box to speed up the process. We first compute the linear boundary from the resized foreground image, and then apply the interpolated boundary to the original image.

3.2.3 Experimental Results

To evaluate our method, we compose a dataset which includes 81 human-shadow images from UCF YouTube Action Dataset [52]. As shown in Figure 3.5, the selected images are the single frames from 27 YouTube hu-

man action videos. We extract 3 nonconsecutive frames from each video. The foreground mask of each image is manually segmented into *shadow* and *non-shadow* subregions. We perform two types of experiments to show the detailed performance of the proposed descriptor and the accuracy of our method as a whole. We randomly divide the videos into two parts, which contain 14 and 13 videos respectively. We use all the 42 images from the 14-video part for training and the rest of the 39 images for testing.

In the first experiment, we compare the ROC curve of the proposed shadow descriptor with those of the reduced feature descriptors. The 4 descriptors in comparison are HSI-$\phi\lambda$-HOG, $\phi\lambda$-HOG, HOG-HSI, and HSI-$\phi\lambda$, where $\phi\lambda$ represents the modified log-polar coordinates. We evaluate the performance by computing their area under the ROC curves (AUC) in Figure 3.4. As expected, the proposed descriptor (solid line) contains all the features and outperforms others. To measure the contribution from each feature, we use the AUC difference between the full feature descriptor and the reduced feature descriptor as a measurement. For example, the importance of HOG feature is measured by the AUC difference between the red and black curve. We are surprised to find that the HSI color feature contributes the least to the detection. The pixel location in modified log-polar coordinates is the most discriminative feature for shadow detection. For the second experiment, we average the per image detection accuracy over 5 rounds of random video partitions. The average accuracy is *96.37%*. Moreover, we measure the accuracy improvement by imposing the spatial constraint. That is, we compare the detection accu-

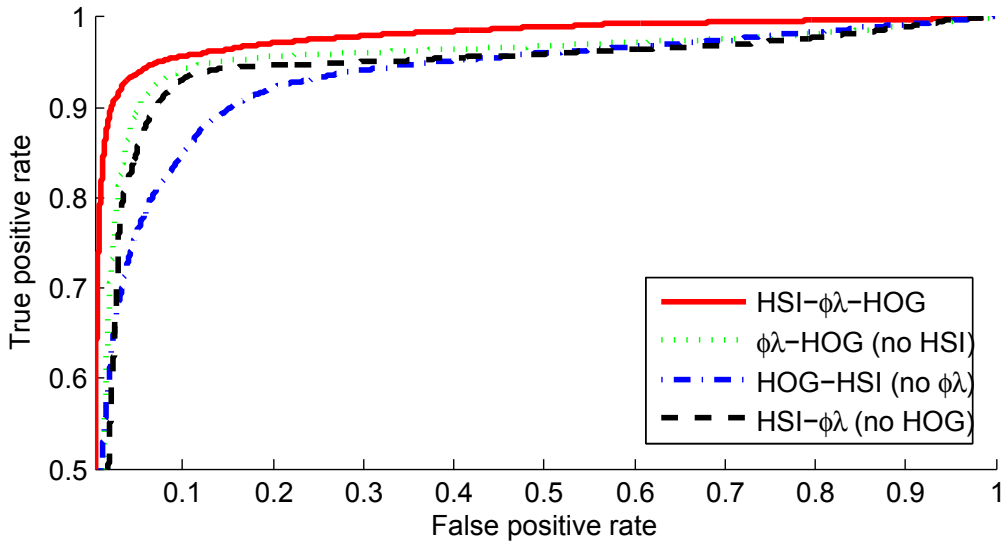


Figure 3.4: ROC curves of the proposed (solid line) and the reduced feature shadow descriptors (dashed lines).

racy before and after the line fitting. The average accuracy improvement per image is *1.3%*, while the accuracy improvement is brought to *84.6%* of the testing images. In MATLAB® implementation on a Pentium D 2.8GHz PC, the average time required to process a 10,000-pixel foreground blob is about 3 seconds. Figure 3.5 demonstrates the qualitative results on regular resolution imagery. We show 9 sets of the processing sequence. Furthermore, we have also processed several tracks of human objects from the VIRAT Aerial Video dataset [58] using the summation of the square of consecutive image differences as foreground mask. Figure 3.6 shows our representative results on this low-resolution dataset.

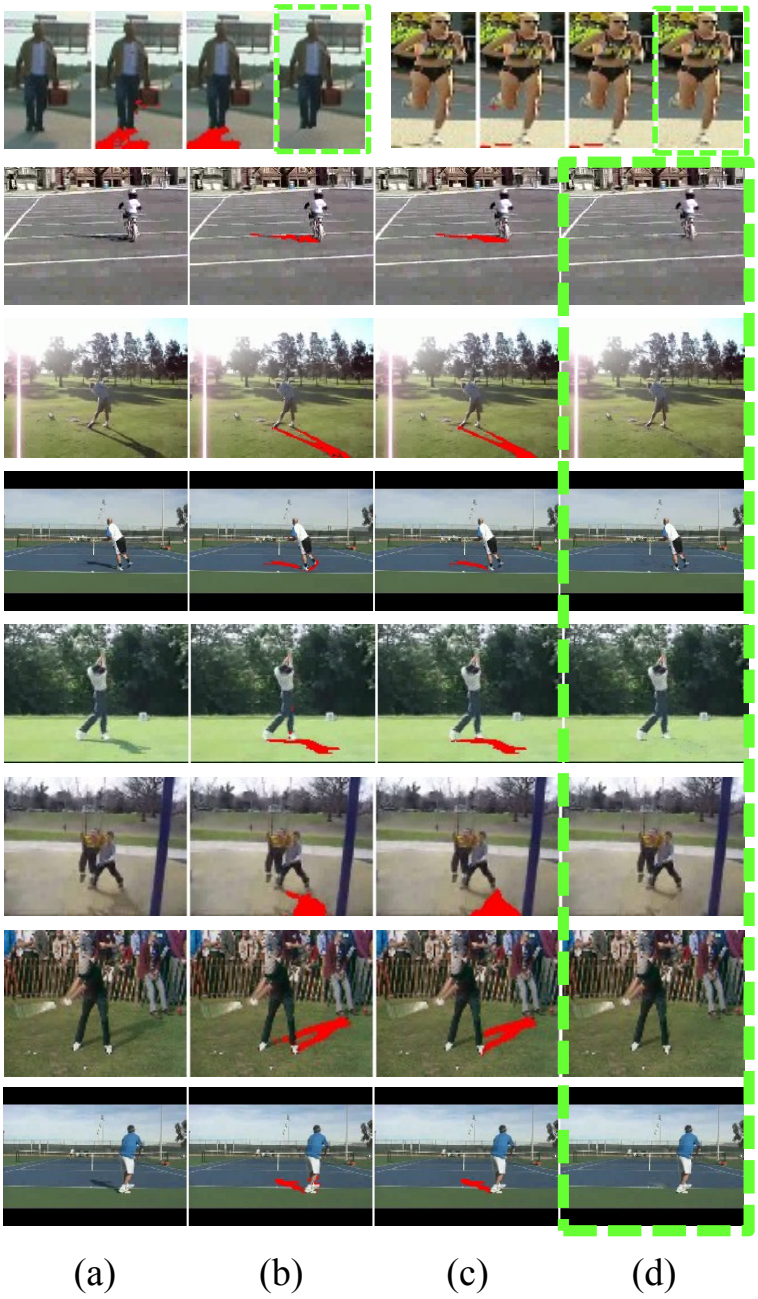


Figure 3.5: Processing sequences of our method. The images in each sequence correspond to (a) the original, (b) detected pixels marked, (c) detected region marked, and (d) shadow removed image.

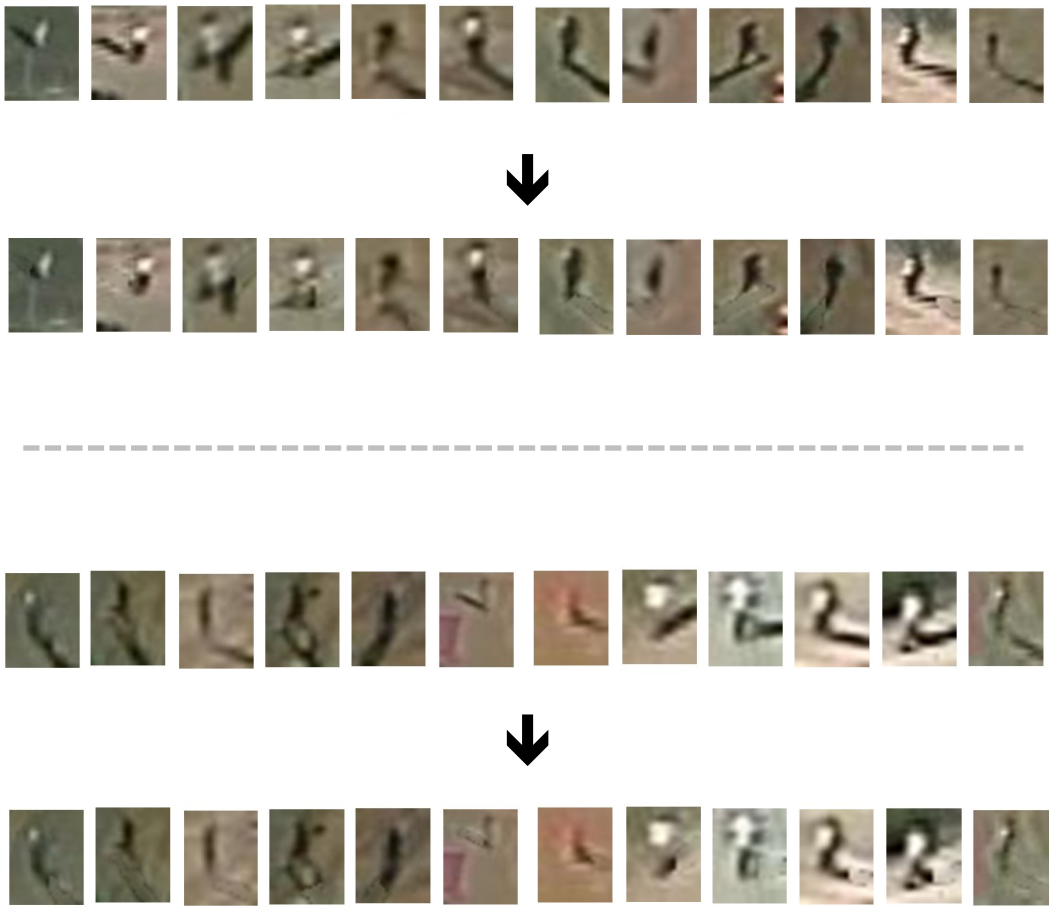


Figure 3.6: The qualitative results of our shadow removal technique on the selected tracks of VIRAT Aerial Video dataset [58].

3.2.4 Conclusions

We present an effective technique to remove human shadows. The major contribution of this work is two-fold. First, we propose a multi-cue shadow descriptor which provides more reliable characterization of shadows. Our shadow detector is able to achieve high accuracy, although the classifier is trained and tested on images from different sets of videos. Second, the proposed 3-stage process largely reduces the risk of classifying human pixels as *shadow* and leaves the shadow removed figure as an intact region. Our method has led to accurate recognition of activities in [15].

Chapter 4

Human Action Recognition by Combining Discriminative Low-level Features

Our goal is to recognize actions from video sequences where human figures are 20 to 50 pixels in height. This is usually the case when actions are being imaged from a far field of view. Therefore, not only is the image resolution greatly reduced, but also the quality of visual cues is adversely effected due to turbulence. As shown in Figure 4.1(a), a person is waving both hands with optical flow vectors superimposed. The average width of his limbs is about 3 pixels, and the boundary between the body parts and background is vague. As a result, the computed optical flow is rather sparse and noisy. In our problem, we find that action classification with a single type of feature is easily subject to background noise and missing features. Moreover, there are certain human actions where one type of feature cannot fully capture their properties. For example, it is difficult to distinguish 'standing' from 'pointing' using optical flow alone. Therefore, instead of describing action by a single type of measure, we propose a novel descriptor which combines both human poses and motion information within a spatial-temporal volume.

We use HOG to represent human poses. The HOG descriptor was

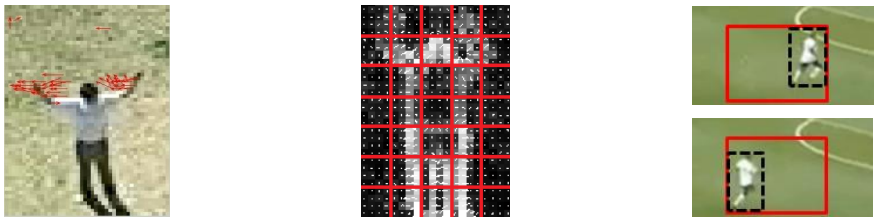


Figure 4.1: (a) Motion feature presented in a far-field of view (b) a human gradient map with our HOG geometry imposed (c) optical flow is computed between the union bounding boxes (red) of two consecutive frames.

originally proposed for human detection [23]. Due to its robustness, similar representations has been successfully applied in the problem of action recognition [38, 51, 54, 74] and object recognition [10]. Similar to the gradient, optical flow is also a directional feature with magnitude. Therefore, we adopt the similar descriptor arrangement of HOG, and characterize human motion by HOF [14, 24].

To synthesize the action descriptor, sequences of HOG and HOF features are extracted from overlapped space-time window of action frames. As in [28], we assume stabilized videos with human tracks are available to us. However, direct concatenation of the time series of both features will end up with a very lengthy descriptor vector. Hence we extend the technique of Supervised Principle Component Analysis (SPCA) [68] to perform feature selection based on the training data. Unlike regular PCA, SPCA aims at selecting a subset of PCs which best separate samples projected from different classes.

The major contribution of this work is two-fold. First we present a compact action descriptor which combines cues of human poses and motion. Our action descriptor is shown to outperform similar descriptors which uses a single type of action feature, applies PCA for dimension reduction, or does not perform SPCA projection. Second, we extend SPCA to perform dimensionality reduction in a multiclass case. This step significantly speeds up the runtime of recognition without sacrificing accuracy. With the combination of RBF kernel SVM, we achieve perfect accuracy on the Weizmann dataset [8] and our own low-resolution dataset called the UT-Tower dataset [65]. For another low-resolution dataset, the Soccer dataset [28], our performance is comparable to other tested methods. This chapter is organized as follows: Section 4.1 introduces the proposed action descriptor including feature computation, feature selection, and action classification. We detail our experimental setup and results in Section 4.2 and conclude in Section 4.3.

4.1 Action Recognition

Our approach for recognizing action from a distant view video is outlined in Figure 4.2. The preprocessing techniques are introduced in Chapter 2. In the following subsections, we briefly review the HOG and HOF action features, which are computed from figure-centric bounding box sequences. Then, we explain the method to select the top discriminative principle components from each feature space. Finally, we present the classification scheme for action recognition.

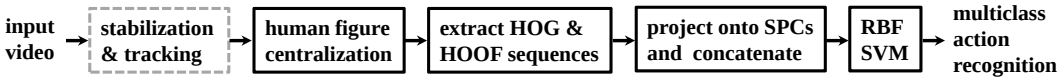


Figure 4.2: Flow diagram of our action recognition scheme. The focus of our method is in solid-line rectangles.

4.1.1 Action Features

HOG. We use the HOG descriptor to characterize details of human poses. The essence of HOG is to describe local edge structure or appearance of object by local distribution of gradients [23]. Without directly using noisy gradient vectors as pixel-wise features, HOG gains robust representation by presenting them as directional patterns over coarser spatial regions.

In HOG implementation, one action frame is divided into non-overlapping spatial grids (cells). For each pixel in the cell, we calculate its gradient vector $\mathbf{g}(x,y) = [g_x(x,y)\ \ g_y(x,y))]^{\mathrm{T}}$. The magnitude and orientation (four-quadrant tangent inverse) of a gradient vector are expressed as

$$m(x,y) = (g_x(x,y)^2 + g_y(x,y)^2)^{\frac{1}{2}} \tag{4.1}$$

$$\theta(x,y) = \tan^{-1}(g_y(x,y)/g_x(x,y)) \tag{4.2}$$

Based on $\theta(x,y)$ and (x,y), every $m(x,y)$ is weighted to vote for the nearest local orientation bins and also the adjacent cell histograms, respectively. Note that $\theta(x,y)$ should be insensitive to the signs of contrasts, because the color variations in clothing and background do not provide extra information for the recognition task. To achieve this, $\theta(x,y)$ is further divided by the modulus π before binning. After accumulating the gradient histogram at each cell,

for better invariance to the illumination changes, the concatenated histogram vector is normalized by the L2-norm. Figure 4.1(b) illustrates our HOG geometry.

HOF. We characterize optical flow by the similar descriptor arrangement of HOG. In addition to the fact that both gradient and optical flow features are measured by 2D vectors, the accuracy of optical flow computation is very susceptible to the quality of image sequence. Therefore, in our scenario, representing optical flow by its local directional distribution is a more reliable option than using it by its exact value.

In the preprocessing steps (see Chapter 2), we have already obtained the accurate estimates of bounding boxes which center on the human figures. Using this information, we are able to locate the minimum rectangular area which covers the moving person between two successive frames. As shown in Figure 4.1(c), the minimum rectangular area is in red and we name it union bounding box. For videos taken from a stationary camera, we compute optical between pairs of successive union bounding boxes. This enable optical flow vectors to encode the figure translation information between frames. The procedure to extract HOF feature is the same as the major steps to compute HOG, except the use of the optical flow feature. We briefly review the important steps and explain the difference.

From the field of optical flow between two union bounding boxes, we extract vectors over the area covered by the first bounding box (dashed box, bottom frame of Figure 4.1(c)). The corresponding optical flow matrix is di-

vided into non-overlapping spatial regions. We measure an optical flow vector by its magnitude $m_{of}(x, y)$ and orientation $\theta_{of}(x, y)$. In a spatial cell, every $m_{of}(x, y)$ is interpolated and aggregated into a local orientation histogram and the histograms nearby. The concatenated motion histogram is normalized to be more invariant to the scale of motion.

Similar to HOG feature, we need to take care of the issue with orientation mapping. In general applications, we do not use directions of actions as a cue to separate them. Therefore, a proper mapping of flow vectors is required so that different directions of the same action are treated as equivalent. The mapping is done by

$$\theta_{of}(x, y) = \begin{cases} \text{sgn}(\theta_{of}(x, y)) \cdot \pi - \theta_{of}(x, y), & |\theta_{of}| > \frac{\pi}{2} \\ \theta_{of}, & \text{otherwise} \end{cases} \tag{4.3}$$

By assuming that the profile view of human actions is being observed, this angular transformation makes motion representation symmetric about the vertical axis. However, there are applications where the direction of action is of interest. For example, in a soccer game, the player's action together with his/her motion of direction is usually considered as a whole. In this case, we can adjust the orientation mapping to meet the requirements accordingly.

4.1.2 Feature Selection and Action Descriptor

Because of the high dimensionality of HOG and HOF features in space-time, we perform dimensionality reduction for each type of the feature vectors before the final concatenation of the descriptor. In general, dimensionality

reduction is carried out by feature extraction and selection. Classical approaches like PCA search for the directions which best represent the sample space. Even though the PCs found by PCA provide an efficient representation of the data, there is no evidence that the projected samples become more separable between classes.

The goal of SPCA is to select a subset of PCs which is most useful for discriminating data projected from different classes. In [68], the task is to detect sources of combustion from infrared imagery. In their binary class problem, PCs are first extracted from positive samples (sources of combustion). To evaluate the capability of a PC to distinguish different classes of data, the discriminative value of a PC is defined as $d = \sigma^+/\sigma^-$, where σ^+ and σ^- are the standard deviation of the projected positive and negative samples, respectively. Therefore, the two classes of data are better separated in the space spanned by PCs with top d.

We extend SPCA to our multiclass action recognition problem. In the feature extraction step, for each action class i, the training samples are divided into $\mathbf{H}^i$ and $\mathbf{H}^{-i}$ according to the labels. Here $\mathbf{H}^i$ denotes a n_f-by-n_i feature matrix where n_f is the length of feature vector and n_i is the number samples from class i. From the autocorrelation matrix of $\mathbf{H}^i$, we extract the matrix of principle components $\mathbf{PC}^i \in \mathbb{R}^{n_f \times n_f}$ by eigen value decomposition. The

discriminative value of the j^{th} component (row) of $\mathbf{PC}^i$ is

$$d_j^i = \sigma_j^i / \sigma_j^{-i} \tag{4.4}$$

$$\sigma_j^i = \sigma(\mathbf{PC}_j^i(\mathbf{H}^i - \bar{\mathbf{H}}^i)) \tag{4.5}$$

$$\sigma_j^{-i} = \sigma(\mathbf{PC}_j^i(\mathbf{H}^{-i} - \bar{\mathbf{H}}^i)) \tag{4.6}$$

and each column of $\bar{\mathbf{H}}^i$ is the mean vector of training samples from class i. In our implementation, we select the subset of PCs, $\mathbf{spc}^i$, of which the discriminative values of components are greater than one. Given a feature vector $\mathbf{h}$, its projection in the new space is

$$\widetilde{\mathbf{h}} = [(\mathbf{spc}^1(\mathbf{h} - \bar{\mathbf{h}}^1))^{\mathrm{T}} \ ... \ (\mathbf{spc}^{n_c}(\mathbf{h} - \bar{\mathbf{h}}^{n_c}))^{\mathrm{T}}]^{\mathrm{T}} \tag{4.7}$$

where $\bar{\mathbf{h}}^i$ is the mean vector of the samples in class i and n_c is the number of total action classes.

To characterize an action sequence, we divide the sequence into overlapped 'chunks' of frames, where each chunk is composed of sequential images of fixed duration. Time series of HOG and HOF features are extracted from every chunk of frames. After projecting them onto the corresponding subspaces, we denote each type of the transformed HOG and HOF vectors by $\widetilde{\mathbf{h}}_g$ and $\widetilde{\mathbf{h}}_{of}$, respectively. The action descriptor extracted from frame $t+1$ to $t+N+1$ (covers N frames of optical flow field) is represented as

$$\mathbf{A} = [\widetilde{\mathbf{h}}_{g;t+1}^{\mathrm{T}} \ ... \ \widetilde{\mathbf{h}}_{g;t+N}^{\mathrm{T}} \ \widetilde{\mathbf{h}}_{of;t+1}^{\mathrm{T}} \ ... \ \widetilde{\mathbf{h}}_{of;t+N}^{\mathrm{T}}]^{\mathrm{T}} \tag{4.8}$$

which is further normalized by L2-norm before being employed by the classifier.

4.1.3 Action Classification

To perform action classification, a multiclass SVM classifier is trained with labeled action descriptors. We adopt the implementation [13], of which the classifier prediction is made by a collection of one-against-one SVM classifiers. In the training phase, each binary SVM classifier leads to an inequality constrained quadratic optimization problem. We choose RBF kernel for our SVM classifier because of the nonlinear relation between action classes and histogram features.

To estimate the best classifier for a dataset, grid search is performed in the space of parameter C and γ, where C is the weight of error penalty and γ determines the width of RBF kernel. The SVM classifier is decided by the set of (C, γ) which maximizes the cross-validation rate in the space of search. In the test phase, a preprocessed action sequence is segmented into intersected chunks of frames, where each chunk is characterized by an action descriptor. After SVM classification, descriptors are evaluated by the probability estimates of actions. We accumulate the probabilities over component descriptors, and classify the sequence as the action which gains the maximum votes.

4.2 Experimental Results

We have tested our method on three datasets, which include the normal resolution Weizmann dataset [8], the low-resolution Soccer dataset [28], and the low-resolution UT-Tower dataset [65]. We evaluate the performance on each dataset by leave-one-out cross validation, where one single action sequence

is selected for testing at a time.

In general, good recognition results are achieved by setting the side of spatial cell to be the width of human limbs. The resolution of orientation bins is ranged from 10° to 20° depending on the dataset. To ensure the distribution of optical flow is not too sparse, we reduce the frame rate by half. Each chunk of frames covers 5 frames, and overlaps with the previous chunk by 4 frames.

Weizmann dataset. The Weizmann human action dataset contains 10 types of human actions performed by 9 different people. Every action is repeated 9 to 10 times so that there are 93 sequences in this dataset. The snapshots of action categories are shown in Figure 4.3(a). We use the provided foreground masks to extract human figures with fixed aspect ratio. Our method achieves 100% accuracy on this dataset. We list other reported results in Table 4.1 as a comparison.

Soccer dataset. The Soccer dataset is a low-resolution dataset collected by Efros *et al.* [28] from several minutes of World Cup soccer game. This dataset contains 66 action sequences from 8 classes. As shown in Figure 4.3(b), actions are distinguished by both action categories and the proceeding directions. Due to the high confusion between 'walk in/out' and 'run in/out', we treat them as the same action as in [29]. We also change the orientation mapping so that the in and out directions of the same action are recognized as the mirror of each other. Our performance and other reported per-descriptor accuracy on each action are presented in Table 4.2.

Besides the low-resolution video frames, the Soccer dataset poses other

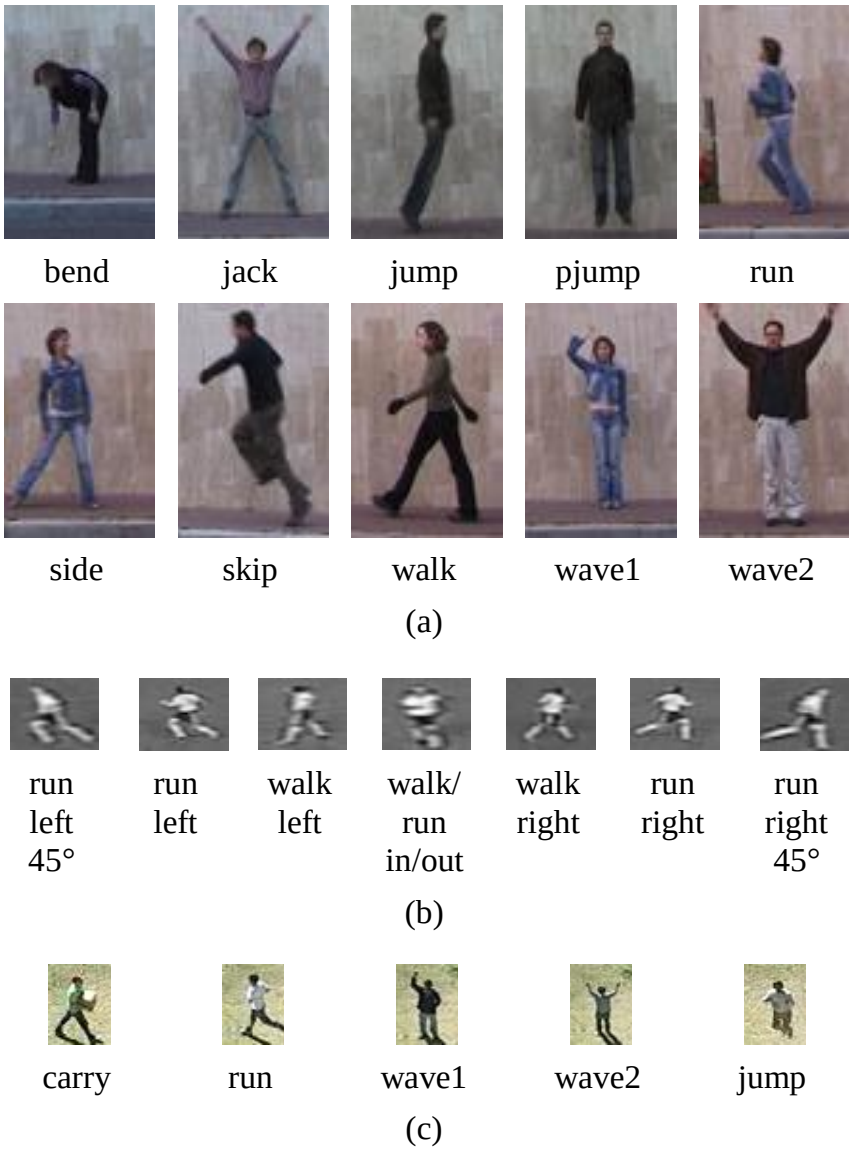


Figure 4.3: Sample frames from each action of (a) Weizmann dataset (b) Soccer dataset (c) UT-Tower dataset.

Method	Accuracy (%)
Our method	100
Fathi and Mori [29]	100
Blank *et al.* [8]	99.6
Jhuang *et al.* [44]	98.8
Hutan and Duygulu [38]	92.0

Table 4.1: Reported per-sequence accuracy on the Weizmann dataset.

challenges to the recognition task. For example, in Figure 4.3(b), even a human observer may find it difficult to differentiate between 'run left' and 'run left 45°'. In addition, this dataset provides unstabilized figure-centric frames. Therefore, the computed optical flow does not contain the information of figure translation between frames. The unbalanced number of samples per class also reduces the classification accuracy on the minor classes. To alleviate these problems, we use background subtracted frames and randomly select the same number of descriptors from each class for training.

Except for 'run left/right 45°', our descriptor is comparable or better than other tested methods in Table 4.3. From the confusion matrix, substantial confusion occurs over the pairs of 'run left' versus 'run left 45°' and 'run right' versus 'run right 45°'. We assume that it is because of the nature of histogram representation, and speculate that histogram based descriptors may not be suitable for characterizing the subtle difference between the same type actions with large directional overlap. In most applications, it is expected that the action descriptor is general enough so that, for example, sequences of 'run left 45°' can be represented as the outliers of 'run left' or even 'run' class.

Action	Our Method	Efros [28]	Fathi [29]
run left 45°	0.47	0.67	0.63
run left	0.59	0.58	0.59
walk left	0.78	0.68	0.86
walk/run in/out	0.88	0.85	0.89
walk right	0.81	0.68	0.85
run right	0.58	0.58	0.65
run right 45°	0.52	0.66	0.53
Overall	0.66	0.67	0.71

Table 4.2: Comparison of descriptor level accuracy on each action of the Soccer dataset.

To verify our assumption, we combine the two pairs of actions which cause the most confusion and perform the experiment under the same settings. Table 4.3 shows the descriptor level confusion matrix when the number of classes is reduced to 5. Significant improvement is found over the the combined classes, while minor accuracy reduction is observed from the original actions due do the unbalanced number of samples per class after combination. Based on the class probabilities of the component descriptors of each sequence, the average accuracy per sequence is as high as 82.0%.

UT-Tower dataset. To show the effectiveness of our method on more variety of human actions in low-resolution scenario, we created a dataset where human actions were being filmed from a distance. We name it the UT-Tower dataset[1] because it was taken from the top of a tower building. The UT-Tower dataset contains 60 sequences of 5 different actions performed by 6 individuals. Figure

[1]The UT-Tower dataset contains 5 action categories at the time this paper was written. The updated results on the complete 9-action dataset is shown in Section 5.3.

	run left/ run left 45°	walk left	run/walk in/out	walk right	run right/ run right 45°
run left/ run left 45°	***0.83***	0.08	0.01	0.01	0.07
walk left	0.12	**0.76**	0.09	0.02	0.01
run/walk in/out	0.01	0.07	**0.80**	0.07	0.04
walk right	0.01	0.03	0.09	**0.77**	0.11
run right/ run right 45°	0.04	0.01	0.03	0.12	***0.79***

Table 4.3: The descriptor level confusion matrix of the Soccer dataset when the number of classes is reduced to 5 (the overall accuracy is 78.66%).

4.3(c) shows the sample frames from each action. In this dataset human figures are less than 40 pixels tall; therefore, trained with manually cropped figure-centric patches, the figure centralization detector is applied to ensure that each action frame is well centered on a figure. Following the similar settings of oriented histograms and space-time window, we obtain 100% accuracy on the UT-Tower dataset as well.

To understand the representation effectiveness of different descriptor formats, we illustrate the corresponding ROC curves on downsampled versions of the UT-Tower data. We have tested 4 combinations of action features and dimension reduction methods. They are denoted as PCA-HOG-[], []-HOG-HOF, PCA-HOG-HOF, and SPCA-HOG-HOF. Here PCA-HOG-[] represents the PCA projected HOG descriptor, []-HOG-HOF stands for the full-length

joint feature descriptor, and PCA-HOG-HOF is PCA projected HOG and HOF time series. These descriptors all represent features in a spatio-temporal volume, and are employed by the same SVM classifier (parameters are optimized separately) to perform action recognition.

We perform 3-fold cross validation in a modified way to demonstrate descriptor performance on each action. That is we randomly select 4 sequences from total 12 sequences of each action for testing, and train on the labeled descriptors from the rest of the sequences. For each scale of the image resolution, we show only the ROC curves of action with the least area under the ROC curve (AUC). Figure 4.4(a) illustrates the comparison of all the 4 descriptors in the original resolution. Figure 4.4(b) and 4.4(c) correspond to the descriptor performance when frame resolutions are reduced to 36% and 16% of the original, respectively.

Our action recognition algorithm is implemented with MATLAB® and run on a Pentium 4 2.8GHz PC. Without further optimization, the average time required to classify a 10-descriptor sequence is ranged from 0.2 to 0.5 seconds. However, if we change the descriptor formation by neglecting the SPCA projection step, it takes 1.3 seconds on average. Because of the use of SVM classifier, the run time depends on the number of training samples [72].

4.3 Conclusions

When actions are being observed from a far field of view, available visual cues from human figures are usually sparse and vague. Therefore, action

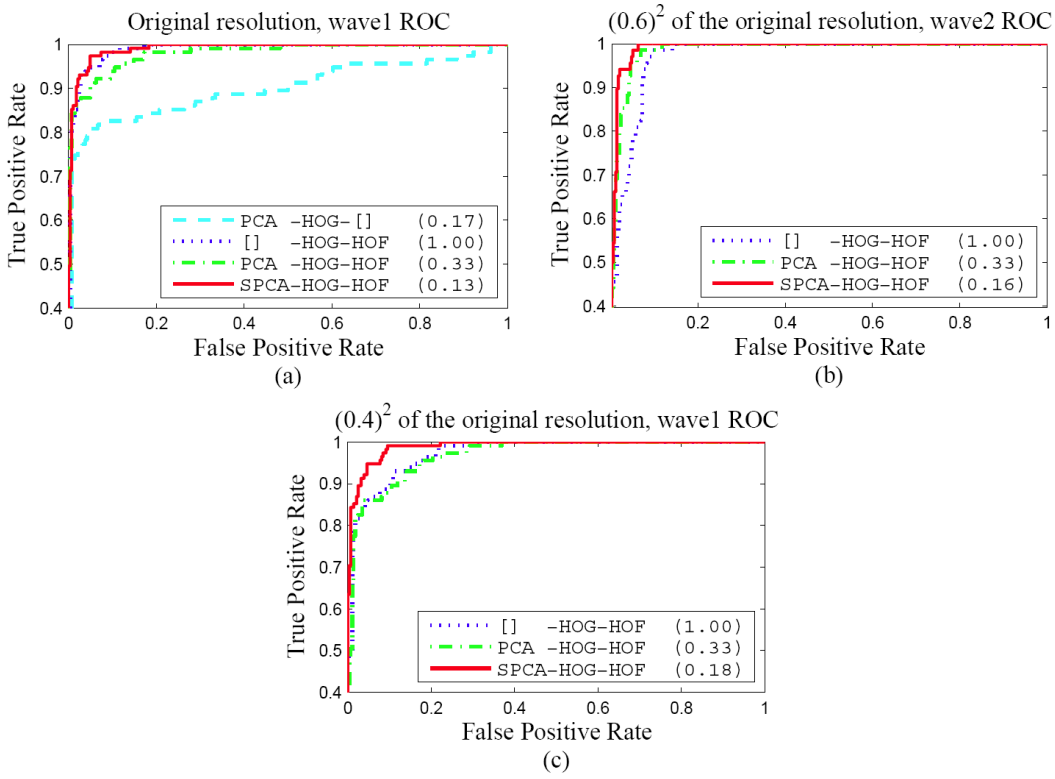


Figure 4.4: For the Tower dataset, we plot the one-against-rest ROC curve for the action with the minimum AUC. The performance of descriptors is evaluated when the frame resolution is (a) original, 40-pixel tall figures (b) 36% of the original, 25-pixel tall figures (c) 16% of the original, 15-pixel tall figures. The decimals in the parentheses represent the ratios of descriptor dimensions to the dimension of a full-length joint feature descriptor. In 4(a), the descriptor does not incorporate HOF feature performs the worst. As shown in (b)(c), the ROC curves of the proposed SPCA-HOG-HOF descriptor occupy the largest AUC in the lower resolution versions of the dataset. Note that as the frame resolution goes down, larger set of **spc** (Eq. (4.7)) is required from each class to provide better separation of projected samples.

recognition algorithms that require an exact description of human shapes or motion may suffer from the quantity as well as the quality of features. The proposed action descriptor is able to better accommodate these issues for two major reasons. First, the use of local orientation histograms to represent features is less susceptible to noisy data. Second, compared to a single feature descriptor, our descriptor is composed of two features so that it is more robust against low quality or loss of features.

Even though a human figure occupies much fewer pixels in a low-resolution video frame, the same amount of feature dimension is still required to characterize an action frame. In particular, our descriptor describes an action as a time series of poses and movements, which take considerable number of dimensions to represent. Moreover, blurry features in low-dimensional imagery add to the difficulty in distinguishing them. To reduce dimensionality while maintaining good accuracy, we extend an existing method to select a subspace of the transformed feature space that provides better separation of projected features for multiple classes.

In our experiments, our method achieves perfect accuracy on both the Weizmann dataset and the UT-Tower dataset. We also show that the proposed action descriptor outperforms other formats of descriptor even when the resolution of figures is reduced to 16% of the original (Figure 4.4(c)). From the results on the Soccer dataset, it is shown that the velocity of the figure as a whole plays an important role in distinguishing directional actions.

Chapter 5

Human Activity Recognition with Mid-Level Features and Speech-Like Processing

In the previous chapter, we propose an action descriptor which is composed of time series of subspace projected pose and motion features. Despite our descriptor provides a comprehensive representation of low-level gradient and optical flow features, the recognition accuracy on real-world low-resolution activity sequences remains significantly lower than that in regular resolution imagery. The reason is because our action descriptor is computed from visual cues that are particularly sparse and noisy. In this chapter, we aim at furthering low-resolution activity recognition performance by employing a novel mid-level feature, which is inspired by a common spectrographic representation of speech. More specifically, our algorithm discriminatively learns action specific local visual patterns and models their occurrence likelihood time series at body parts. This enables us to take advantages of the spectral properties of active parts' movements in addition to local interest features for activity representation.

Human activity recognition and speech recognition appear to be two loosely related research areas. However, on a careful thought, there are sev-

eral analogies between activity and speech signals with regard to the way they are generated, propagated, and perceived. Compared to the research in automatic speech recognition (ASR), human activity recognition is a relatively young discipline. The first ASR system was built in the 1950s [26], and now the commercialized services and products are used in daily lives. These two seemingly unrelated areas share some very similar goals and processing methodologies. For example, we expect an ideal video surveillance system to accurately segment and semantically annotate continuous activities of multiple agents in unconstrained environments. Likewise, the ultimate goal of ASR is to segment and label spontaneous and continuous speech into constituent words then sentences independent of speakers and vocabulary. In addition, activity and speech are both temporal data; therefore, techniques such as Hidden Markov Model (HMM) and Dynamic Time Warping (DTW) are commonly adopted for the recognition of activity and speech sequences.

We are motivated to model human activities as speech due to the analogies between their production mechanisms. While speaking, part of our articulatory apparatus continuously reshape the vocal track which causes time varying resonances of the exhaled air flow. The magnitude of the propagated air pressure wave is a non-stationary signal which is relatively stationary when observed in short time intervals. Therefore, as shown in Figure 5.1, one common way to characterize digitized speech signals is to extract the magnitude spectrum from each equally spaced and overlapped time window (frame in ASR). The representation that concatenates individual spectra in time is called

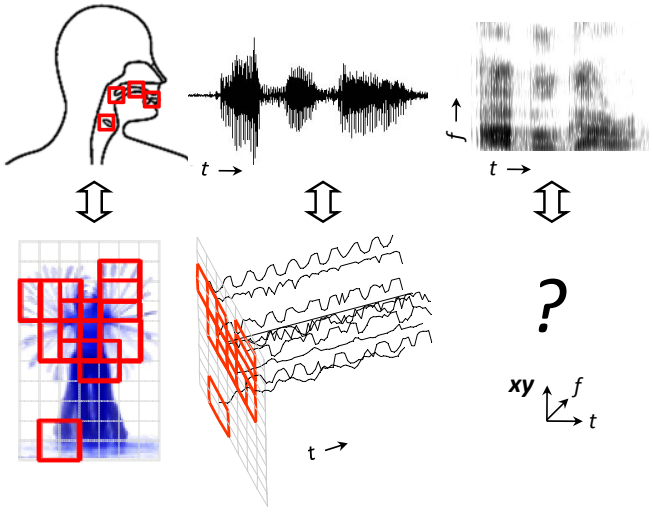


Figure 5.1: We compare human activities to speech, and introduce the analogies between articulatory apparatus and body parts, air pressure wave and local likelihood time series, and spectrogram and our spectrogram-like representation.

a *spectrogram*. The time span of the analysis window is approximately equal to the period while the vocal track sustains its shape (10 to 50ms). This setup validates the assumption that each time segment of the speech signal is quasi-stationary [84].

On the other hand, the motion of human body parts also emit time varying visual patterns at a relatively low frequency band. Nevertheless, if we are to compare body parts to articulatory apparatus, there are two minor differences to be clarified. First, it is mainly the shape of the articulatory apparatus that manipulate the articulation of phonemes, while human actions are distinguished by the simultaneous interest patterns (of motion or gesture) from parts. Second, for speech, the waveform of sound is already synthesized

within the vocal track, while in action different body parts create different visual patterns and are perceived as a whole. In the ASR community, there has been an emerging interest in incorporating visual information for recognition. Lips are the most visible articulatory apparatus; therefore, various visual features [20] extracted from the corresponding area have been shown to further recognition performance. Despite all the "acting apparatus" being directly visible, there is little in the way of research exploring the temporal signals [5, 56] emitted from body parts for activity recognition (the sounds of actions).

Similar to speech signals, if we are able to model the associated interest patterns of an action at body part level, their occurrence likelihood in a short time period can be also deemed as quasi-stationary. Based on this observation, we propose a spectrogram-like representation to characterize human activities. We name it *action spectrogram* (AS). Compared to a 2D spectrogram, AS is a space-time-frequency representation which records the occurrence likelihood spectra of action specific interest patterns emitted from body parts. However, there are three major issues to be solved to make this kind of representation possible:

- how are local interest patterns defined and located?
- how are local interest patterns associated with actions?
- how do we model the occurrence time series of local interest patterns?

In this work, we provide a complete solution to these issues. First, we define

local interest patterns as the video content indicated by the spatio-temporal interest points (STIP) [27] within a figure-centric action sequence. Second, to associate local interest patterns with actions, we modify Adaboost algorithm to learn a set of action associated spatio-temporal interest point detectors (AASTID) from each action. Third, we use the boosted AASTID to compute the occurrence likelihood of local interest patterns from different body parts. These likelihood time series are divided into overlapped short time segments (likelihood segments) and converted by an 1D Fast Fourier Transformation (FFT) to synthesize AS. We train SVMs to classify an activity AS into the component actions.

Our work provides a novel perspective to the characterization of human activities, which may induce the transfer of research in both areas of speech and activity recognition. We not only make the associations between different aspects of speech and activity signals, but also contribute a viable solution to recognize continuous activities as speech. The remainder of the chapter is organized as follows. Section 5.1 introduces the technical details of AS computation. In Section 5.2, we present the methodologies to classify a single AS slice and a continuous AS sequence as a stream of activity. We demonstrate our experimental results on 4 diverse datasets in Session 5.3, and conclude in Session 5.4.

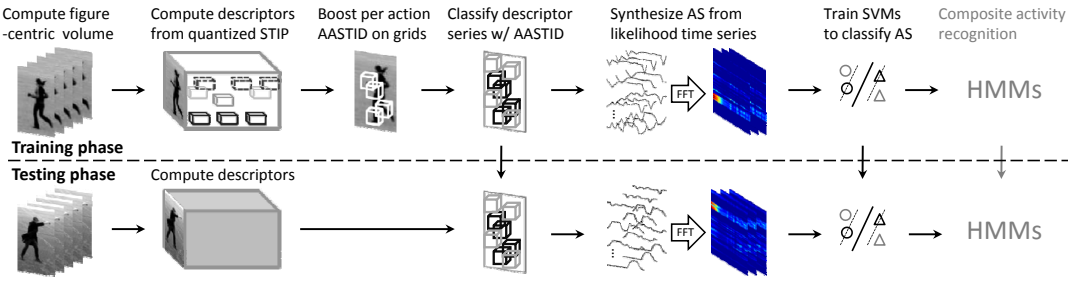


Figure 5.2: Flow diagram of our activity recognition scheme. The vertical arrows indicate the supply of trained models.

5.1 Action Spectrogram

An important process in the computation of AS is to quantize the occurrence time series of action specific local interest patterns. This involves the stabilization of human figures, the learning of action associated local patterns, and the estimation of occurrence likelihood. Also, we need to evaluate the proper time interval to divide continuous likelihood series into short segments, which are synthesized into AS. This process is a part of the overall algorithm as in Figure 5.2.

5.1.1 Preprocessing and Action Features

Preprocessing. Our algorithm, similar to [28, 29], operates on the figure-centric spatio-temporal volume of activity. Depending on the setup of the activity recognition system, this generally requires detection and continuous tracking of human objects. In most of our tested datasets [8, 65, 69], there is one human object in a frame; therefore, we perform tracking by human detection. The details of our preprocessing algorithms are presented in Chapter

3.

Action Features. We use both shape and motion histogram based features to characterize human activities. In addition to the performance benefits, combining features of different types provides a broader coverage of activities. For example, there are scarce features due to motion which can be extracted to distinguish certain static actions such as 'stand' from 'gesture'. More specifically, within a figure-centric volume, we represent successive poses by time series of HOG and motions by time series of HOF.

HOG descriptor divides the subject figure into equally spaced regions called cells, and represents the edge structure of each cell by the angular distribution of gradients. Compared to a cell, a block covers a larger region which consists of several cells. In [23] the cell histograms within a block are normalized to provide better invariance to illumination and shading. Here we characterize the appearance of human body parts at the spatial scale of a block. The overlapped blocks are more robust against minor stabilization errors and describe parts with the context of adjacent cells. As shown in Figure 5.3(a), our implementation uses 2×2 cell blocks and follows the common settings as in [23]. Note that we compute HOG time series via figure stabilization.

We describe the motion field between each pair of successive figure-centric frames by HOF descriptor. Besides the types of feature being characterized, the main difference between HOG and our modified HOF descriptor is the orientation mapping carried out. We follow [23] to use unsigned gradient vectors in HOG computation. In general applications, the acting directions

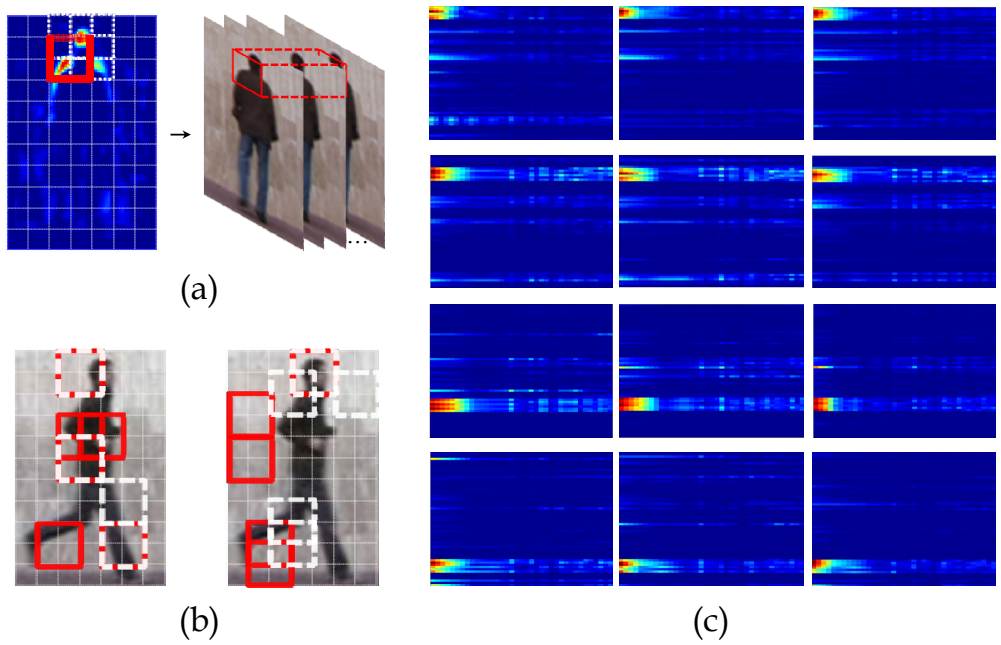


Figure 5.3: (a) Left: a slice of a STIP response volume. By referring to it, we quantize a local maximum at the head position to a grid location. (b) Left: D_{run} boosted from quantized STIP as in (a). Right: D_{run} boosted from dense video grids. The solid squares are gradient based D_{run}, and the dashed ones are optical flow based. The D_{α} computed by our method effectively capture the action associated body parts instead of some random background. (c) The sample AS time slices from the sequences (columns) of different actions (bend, jack, walk, wave1 in row) from [8].

of a person are not used as a cue to distinguish actions. Therefore, to make optical flow vectors symmetric about the vertical axis, the orientation of a flow vector is converted by

$$\theta_{of} = \begin{cases} sgn(\theta_{of}) \cdot \pi - \theta_{of}, & |\theta_{of}| > \frac{\pi}{2} \\ \theta_{of}, & otherwise. \end{cases} \tag{5.1}$$

The cells of the HOF descriptor capture the relative motions of parts at a finer spatial scale. The same as HOG, we describe the motion patterns of parts in every 2×2 cell block.

5.1.2 Learning Action Associated Interest Patterns

We compare the learning of action associated local interest patterns to the search of correspondence between a uttered phoneme and the shapes of those active articulatory apparatus. Actions appear to vary across both time and body parts; however, not every local video feature contributes to the correct recognition of actions. One effective technique to select a variety of discriminant features is to evaluate the weak classifiers trained on an over-complete set of features [78]. Our method is similar to Ke *et al.* [46] and Laptev and Pérez [48] in the sense of discovering discriminant cubic features in a boosting framework. Nevertheless, our work differs from theirs with regard to the method of selecting boosting instances and the format of action classifiers as suggested in Section 2.3 and to be detailed later.

In general, STIP detectors are used to localize local video structures which pose significant variations in both space and time. Our AASTID are

boosted space-time window classifiers, which are *not* trained to detect points of interest but to produce the occurrence likelihood of action specific STIP. Here we assume, within the figure-centric volumes of the same action, the STIP that are in close *spatial* proximity of each other present similar interest patterns. One important observation that motivates us to boost AASTID from STIP is that action associated interest patterns occur in an intermittent fashion. For example, in a spatio-temporal volume of a person 'kicking', the most descriptive video cuboids cover the sweeping leg in time. However, after the leg goes down, there is no subsequent interest pattern emitted from the leg position until the next kick. Previous methods such as [46, 48] select boosting examples by randomly sampling cubiods from dense video grids. Their approach inevitably includes positive features from video cuboids which do not relate to the action (*e.g.* arbitrary background volumes) and negative features which do not characterize the rest of the actions well. As a result, the discriminating power of the boosted weak classifiers are weakened by labeling uninformative video cuboids as positive and negative examples (Figure 5.3(b) for example).

We use the occurrence likelihood series of action associated STIP as features. Ideally the likelihood signal emitted from an AASTID is expected to peak, bottom, and level (about 0.5) when classifying features from positive, negative, and random video cuboids, respectively. We detail the implementation of AASTID as follows.

Extracting STIP. The popular STIP detector proposed by Dollár *et al.* [27] is a combination of a 2D Gaussian spatial kernel and 1D Gabor temporal fil-

ters. Their STIP detector is devised to be responsive not only to periodic motions but also to a wide range of other interesting space-time patterns. Via their implantation, we are able to extract a dense set of STIP to capture the details of a training volume. As shown in Figure 5.3(a), a STIP response volume is computed using $R = (I * g * h_{ev})^2 + (I * g * h_{od})^2$, where $g(x, y; \sigma)$ is a Gaussian smoothing kernel and $h_{ev}(t; \tau, \omega)$ and $h_{od}(t; \tau, \omega)$ are a quadrature pair of Gabor temporal filters. STIP are fired at local maxima by applying non-maximum suppression on the R volume. We quantize a local maxima to the grid location of a 2×2 cell block while maintaining its temporal location. This is achieved by comparing the integrals of R within the quantized video cuboids (section of a block) which overlap with the local maxima. We compute time series of HOG and HOF features from the quantized video cuboid with the maximum R integral. We denote a STIP of action α by $cb_{\alpha}(u, v, t)$, which is characterized by $\mathbf{h}(u, v, t, \theta)$ vectors, where (u, v) is the quantized grid location, t represent the time and the corresponding training volume, and θ indicates the type of histogram feature.

Boosting AASTID. We boost a set of AASTID per action. These detectors are mostly localized at the related body parts (see Figure 5.1). Unlike [48], for reliable estimation of STIP occurrence likelihood, we employ instance weighted linear kernel SVM [13] for weak learners

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \ & \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} C_i\xi_i \\ \text{subject to } & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0. \end{aligned} \tag{5.2}$$

In this primal problem, the inverse of margin width together with the weighted

sum of training errors are being minimized. C_i and ξ_i correspond to the penalty and training error of the instance-label pair $(\mathbf{x}_i, y_i)$. This SVM formulation enables a weak learner to minimize the classification error of samples weighted by previous boosting iterations. Our weighted SVM based weak classifiers are more robust than those of weighted Fisher Linear Discriminant [48] based given the limited number of training instances.

We modify Adaboost to learn AASTID from spatio-temporally scattered STIP. We follow the basic settings as in [78] and focus on presenting the differences. The set of AASTID boosted from action α are among the best weak learners D_α of the total $(nr-1)\times(nc-1)\times nf$ weak learners d_α, where nr, nc, and nf are the numbers of cell rows, columns, and feature types. For each grid location (u, v), we denote the set of all STIP time instances as $T(u, v)$. A weak classifier $d_\alpha(u, v, \theta)$ is learned to distinguish θ represented $cb_\alpha(u, v, T_\alpha)$ from $cb_{\neg\alpha}(u, v, T_{\neg\alpha})$, where $T_\alpha \cup T_{\neg\alpha} = T(u, v)$ and $T_\alpha \cap T_{\neg\alpha} = \emptyset$. The weighting of $\mathbf{h}(u, v, t, \theta)$ at iteration i is $w_i(u, v, t, \theta)$, which is updated by intersecting t with $T(u_{best}, v_{best})$ of iteration $i-1$ $(i > 1)$. $w_i(u, v, t, \theta)$ is updated to $\frac{\epsilon_{i-1}}{1-\epsilon_{i-1}} w_{i-1}(u, v, t, \theta)$ if and only if $cb(u, v, t)$ is only temporally overlapped with the correctly classified $cb(u_{best}, v_{best}, T_{correct})$ where $T_{correct} \subset T(u_{best}, v_{best})$ and ϵ_{i-1} is the minimum weighted error in $i-1$. This implies that any $cb(u, v, t)$ overlap with the wrongly detected $cb(u_{best}, v_{best}, T_{wrong})$ or missing a temporal intersection will be emphasized in the next iteration.

Estimating likelihood. Similar to the preprocess of speech signals, our weak learners are trained to output calibrated likelihood values. Given the

histogram vector, $\mathbf{h} \in R^n$, and the indicator of α, $y \in \{0, 1\}$, we aim to estimate the posterior probability $p(y = 1|\mathbf{h})$ using D_α. The method proposed by Wu *et al.* [85] approximates the posterior probabilities by coupling them with pairwise class probabilities. They start with modeling each pairwise class probability as a sigmoid of the corresponding decision value f

$$p(y = i|y = i \cup j, \mathbf{h}) \approx \frac{1}{e^{Af+B}}, i \neq j \tag{5.3}$$

where A and B are obtained by minimizing the negative log-likelihood function while f is calculated by performing cross-validation on the training set. The formulation of their pairwise coupling is based on the Bayesian equality

$$p(y = i|y = i \cup j, \mathbf{h})p(y = j|\mathbf{h}) \\ = p(y = j|y = i \cup j, \mathbf{h})p(y = i|\mathbf{h}). \tag{5.4}$$

This equality simply suggests that $p(y = i|\mathbf{h})$ is proportional to $p(y = i|y = i \cup j, \mathbf{h})$ in a binary problem, while it requires convex optimization for a multi-class problem.

5.1.3 Synthesizing Action Spectrogram

Compared to a sound spectrogram, the additional dimension of space in our representation characterizes the spatially distributed AASTID. We classify a figure-centric action volume with the spatial array of AASTID and synthesize the i^{th} AS slice from the frame interval $< (i-1)l_{step} + 1, (i-1)l_{step} + l_D - 1 + l_{seg} >$, where l_{step}, l_D, l_{seg} are the temporal lengths of sampling step,

AASTID, and likelihood segment, respectively. From each $l_D - 1 + l_{seg}$ frame sampled snippet of the volume, we can extract n_D length l_{seg} likelihood segments, where n_D is the total number of AASTID. The likelihood segments of a snippet are transformed by FFT and concatenated along the dimension of space to form a 2D time slice of the AS volume. We show the sample AS slices from 12 sequences of 4 actions in Figure 5.3(c), where one action is distinguished not only by the active AASTID responses (bright rows) but also by its *spectral signature* (bright columns). For effective characterization of action, the selection of AASTID and the estimation of l_{seg} are introduced.

Selecting AASTID. As we boost the best weak learners on the spatial grids, they represent the most valid weak hypotheses about the action in the measure of detection rate; however, it is their spectral waveforms that are directly used as features. Therefore, we trim the best weak classifiers of each action to form the contributed set of AASTID. Let $D_\alpha(i)$ be the i^{th} best weak classifier of α, where i represents the trippet of (u_i, v_i, θ_i). We classify both the positive and negative $(\neg\alpha)$ training volumes with $D_\alpha(i)$ and divide the emitted likelihood time series into n^+ and n^- fixed length segments. The spectra of the segments are denoted as $\{\mathbf{x}^+_{1,i}, \mathbf{x}^+_{2,i}, ..., \mathbf{x}^+_{n^+,i}\}$ and $\{\mathbf{x}^-_{1,i}, \mathbf{x}^-_{2,i}, ..., \mathbf{x}^-_{n^-,i}\}$. The discriminative value, $F(i)$, of $D_\alpha(i)$ emitted spectra is formulated as a Fisher discriminant like score [19]

$$\frac{\left\|\bar{\mathbf{x}}^+_i - \bar{\mathbf{x}}_i\right\| + \left\|\bar{\mathbf{x}}^-_i - \bar{\mathbf{x}}_i\right\|}{\frac{1}{n^+-1}\sum_{j=1}^{n^+}\left\|\mathbf{x}^+_{j,i} - \bar{\mathbf{x}}^+_i\right\| + \frac{1}{n^--1}\sum_{j=1}^{n^-}\left\|\mathbf{x}^-_{j,i} - \bar{\mathbf{x}}^-_i\right\|} \tag{5.5}$$

where $\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_i^+$, and $\bar{\mathbf{x}}_i^-$ are the mean spectra of the entire, positive, negative training sets. The D_α with top F values are selected as the contributed set of AASTID from α.

Estimating l_{seg}. One popular approach to analyzing activities is to divide a video into snippets of frames and perform recognition from the snippets. In most of the literature, the duration of individual snippets is decided heuristically. Our speech-like representation of action provides a ready medium to tackle this problem. That is, by assuming each action is a random process, we can approximate the proper l_{seg} by performing a stationarity test on its realizations (likelihood series). Common methods for the test of stationarity include auto-correlation function and runs test [34]. They all require a sufficient number of samples per realization to make a meaningful judgement; however, most of the dataset videos are shorter than 3 seconds and sampled at a relatively low frame rate. Besides, these tests do not provide a normalized measure to indicate the degree stationarity.

We propose to approximate l_{seg} by calculating the average pairwise spectral similarities over segment lengths. In Figure 5.4, as we sample longer and longer likelihood segments from the same action of [8], the corresponding AS slices converge gradually in waveforms. The average pairwise similarity of the n AS slices of α is computed by

$$S(l, \alpha) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j:j \neq i} NCC(\mathbf{X}_i^\alpha, \mathbf{X}_j^\alpha) \tag{5.6}$$

where $(\mathbf{X}_i^\alpha, \mathbf{X}_j^\alpha)$ represents a pair of flattened 1D AS slices synthesized from

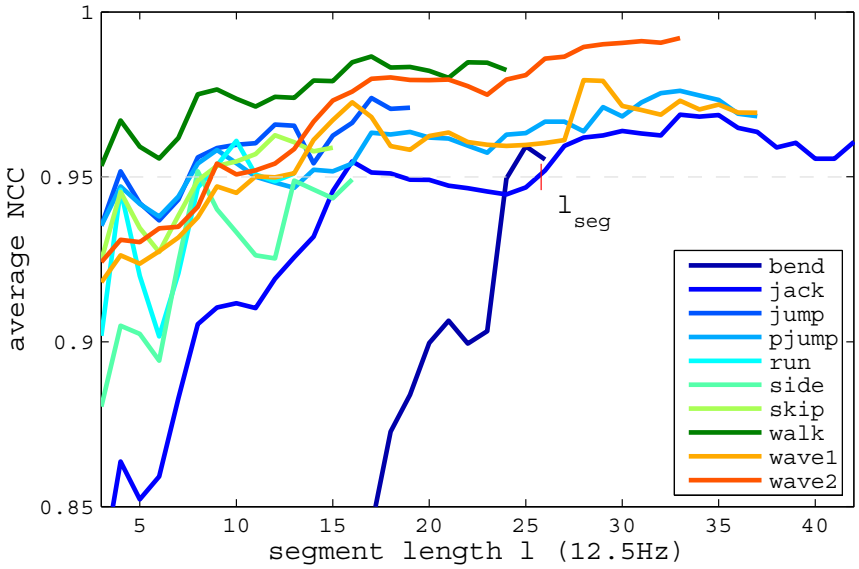


Figure 5.4: The average spectral similarities of AS as functions of l, which are used to determine l_{seg}. The likelihood segments are sampled with less than $\frac{1}{2}$ temporal overlap. The length of a curve depends on the duration of its longest sequence.

length l segments, and NCC is short for Normalized Cross-Correlation. Given the target correlation value, we approximate a sufficient segment length, l_{seg}, by thresholding the similarity curves. The AS of aperiodic actions such as 'bend' require longer l to capture the complete occurrence. Note that we can certainly use a large l_{seg} to meet the target correlation value; however, this inevitably reduces the time resolution of the recognized activity.

5.2 Classification

We train a collection of one-against-one linear SVM classifiers [13] to recognize the AS slices of different actions. We prefer linear SVMs to other

linear classifiers because they are rather discriminant while providing better out-of-sample generalization. Moreover, compared to nonlinear classifiers, they are easy to train, fast to run, and achieve consistently decent performance on different datasets and feature settings. We have tested several nonlinear kernel SVMs on our spectral data, for example, RBF, multi-channel Gaussian [89], and NCC kernel [79]. In our experiments, these nonlinear SVMs usually perform similarly or slightly better than the linear ones; however, their testing accuracies are sometimes subject to overfitting.

To recognize composite human activities, we consider a hybrid HMM approach [32], which has been implemented for real-world ASR applications. Traditional HMM based ASR systems model the state emission probabilities of phonemes using mixtures of Gaussians, which are replaced by more sophisticated classifiers such as Artificial Neural Networks (ANN) or SVM in a hybrid system. With the states corresponding to phonemes, spoken words are modeled by individual HMMs. Likewise, we slice-wise classify an activity AS into a sequence of actions, and model the temporal evolution of the sequential actions via activity HMMs. Our linear SVMs are trained to output the posterior probabilities (see Equation (5.3) and (5.4)) of actions, which can be applied to activity HMMs. Note that using our representation, interaction types of activities can be modeled by specialized HMM; for example, Oliver *et al.*[59] use coupled HMM to recognize person-person interactions.

5.3 Experimental Results

Figure 5.5 summarizes the 4 datasets adopted to evaluate our method. The challenges posed by these datasets include low-resolution, blurry imagery, shadows, broken tracks, and variations in viewpoints, scales, scenes, lighting conditions, and clothing. We follow the same principles to initialize the parameters across datasets. For the computation of orientation histograms, we use 9 bin histograms, and set the block size approximately equal to $\frac{2}{3}$ of the limb length with the stride (block overlap) of a cell size. The histogram time series extracted from a video cuboid is normalized with L2-Hys [23]. We manipulate the values of σ and τ so that there are about 20 to 50 STIP fired per second depending on the complexity of the training action. Based on Equation (5.5), no more than 10 AASTID per action are selected among the best weak learners which score less than a 45% error rate. To speedup runtime, we reduce the video frame rate to half of the original, but double the time resolution of likelihood series by spline interpolation. $2^{\lceil \log_2^{l_{seg}} \rceil}$-point FFT is performed to synthesize AS. For videos shorter than l_{seg}, we replicate the existing likelihood series so that there is at least one AS slice formed per sequence. Finally, we compare the proposed algorithm in this chapter with the space-time joint feature descriptor presented in Chapter 4 on the same 4 datasets. Our experimental results and findings are detailed as follows. **Weizmann.** The Weizmann dataset [8] was filmed at medium resolution in a controlled environment. This dataset consists of 93 sequences of 10 actions performed by 9 individuals. We apply our preprocessing technique to extract figure-centric volumes, because

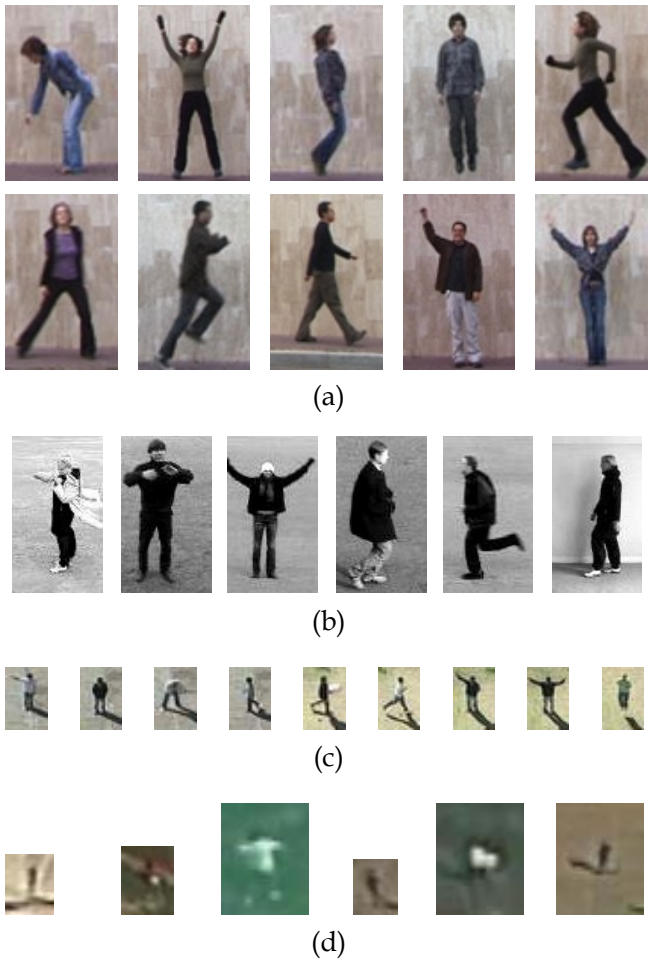


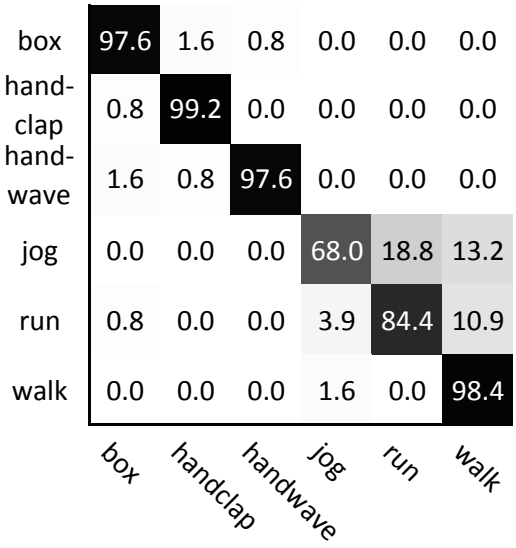
Figure 5.5: We tested our method on 4 datasets: (a) Weizmann (b) KTH (c) UT-Tower (d) VIRAT Aerial Video. The actions are self-explanatory from the figures except those from the Aerial dataset, where the actions are 'stand','dig','throw','walk','carry', and 'run'.

some of the provided foreground masks contain incomplete figures (e.g. shahar_side). Evaluated with leave-one-sequence-out cross-validation (LOOCV), our method achieves 100% accuracy on this dataset.

KTH. Similar to the resolution setting of Weizmann, KTH [69] is a much more challenging dataset. As shown in Figure 5.5(b), KTH is comprised of 6 actions, which were taken at varied scales with persons wearing different clothing in different scenes. The entire dataset contains 2,391 short clips acted by 25 individuals. We follow the setup as in [69] to partition the dataset into 3 parts by person identity. We use $\frac{2}{3}$ of the dataset for training and the other $\frac{1}{3}$ for testing. Our linear SVMs correctly recognize 94.4% of the AS slices in the testing set. The average accuracy per action is 90.9%. The confusion matrix together with the comparison with other reported methods are tabulated in Table 5.1.

We are surprised to find that the per-video accuracy is about 3.6% lower than the per AS slice accuracy (90.8% *v.s.* 94.4%). After examining the error sequences, it is discovered that a significant portion of the misclassified clips are shorter than l_{seg} (1.5 seconds); however, these short clips represent 27% of the test set. Therefore, we conjecture that the disturbing likelihood spectra caused by an insufficient number of samples ($< l_{seg}$) and padding artifacts have led to the high error rate in short clips.

UT-Tower. The UT-Tower dataset [15] is a low-resolution dataset where actions were filmed top-down in a near aerial view and the human figures are 20 to 30 pixels in height. This dataset is composed of 9 actions performed



Method	ACC %
Ryoo [64]	91.1
Proposed	**90.9**
Fathi [29]	90.5
Niebles [57]	83.3
Dollár [27]	80.2
Schuldt [69]	71.7
Ke [46]	63.0

Table 5.1: Our results on the KTH dataset: the confusion matrix for per-video classification and the comparison with other methods.

by 6 persons in 2 scenes. Each subject repeats the same action twice so that there are 108 sequences. We perform LOOCV to compare with other methods as in [65]. The accuracy of our method is 98.2%, which is the best result reported on this dataset so far. The two incorrectly classified sequences are the 9^{th} sequence of 'walk' and the 5^{th} sequence of 'wave2', in which the low color contrast between a person's clothes and background confuses the classifier.

VIRAT Aerial Video. For the previous 3 datasets, our speech-like representation and recognition strategy demonstrate results that are better than or comparable to the state-of-the-art. To test the effectiveness of our methodology, we challenge it with video sequences taken from a Unmanned Aerial Vehicle (UAV). We manually select 42 sequences out of 6 actions from a large collection of UAV recorded footage named the VIRAT Aerial Video dataset

	stand		dig		throw		walk		carry		run	
stand	50.0	37.5	0.0	12.5	0.0	0.0	50.0	37.5	0.0	0.0	0.0	12.5
dig	0.0	12.5	37.5	37.5	0.0	0.0	37.5	12.5	12.5	25.0	12.5	12.5
throw	0.0	0.0	0.0	20.0	40.0	20.0	20.0	20.0	20.0	40.0	20.0	0.0
walk	12.5	25.0	12.5	0.0	0.0	12.5	37.5	12.5	37.5	50.0	0.0	0.0
carry	0.0	0.0	12.5	25.0	25.0	0.0	25.0	12.5	25.0	37.5	12.5	25.0
run	0.0	0.0	20.0	20.0	20.0	20.0	20.0	0.0	0.0	0.0	40.0	60.0

Table 5.2: The confusion matrices of ours (AS) and a baseline method (HOG time series) on the selected VIRAT Aerial Video dataset. The pair of percentages in each bi-colored cell represent our/baseline accuracy. The overall accuracies are 38.3% *v.s.* 33.3%.

[58]. The resolution of the videos is 720×480 pixels with the tracks of objects computed at 10 fps. As shown in Figure 5.5(d), the imagery taken from an UAV not only creates difficulties due to low figure resolution, but also poses problems with vague object appearances, salient shadows, interrupted tracking (person temporarily out of FOV), and time varying viewpoints and scales. Due to these issues, part of the footage even requires repeated human scrutiny to perform ground truth annotation. Therefore, to propose a meaningful evaluation set, we select tracks of human actions which do not require a second inspection for labeling.

We refine the tracks with the preprocessing step to acquire stabilized action sequences. Even with this additional process, the quality of the extracted bounding boxes cannot be as consistent as those acquired from the

other 3 datasets (see Figure 5.5). To assess the performance of our method, we compare our accuracy with that of a baseline approach. We adopt time series of HOG extracted from overlapped spatio-temporal volumes (match the AS computation intervals) as the baseline descriptor. For the sake of fair comparison, we train linear SVMs on the HOG descriptors and use LOOCV accuracy as a measure. The average accuracy of our method is 38.3%, while it is 33.3% for the baseline approach. The confusion matrices are summarized in Table 5.2.

Comparison. We have shown results of the proposed representation and recognition scheme on the 4 public datasets. To quantify the possible improvements our spectral characterization of activity brings about, we repeat the experiments on the 4 datasets using the algorithm described in Chapter 4. Table 5.3 summarizes the dataset specifications, evaluation protocols, and recognition accuracy. In this table, SPCA-HOG-HOF represents the subspace (via SPCA) projected space-time HOG and HOF descriptor. Results show that the proposed algorithm improves our previous recognition approach by 2 to 4 percent on the low-resolution datasets.

5.4 Conclusions

We have presented a novel activity recognition scheme which adapts naturally from ASR. We use both local video content and occurrence likelihood spectra to verify actions. More specifically, localized at body parts, the AASTID are trained to be responsive only to action specific interest patterns.

	Dataset	# Actions / # Videos	Resolution (pixels)	Evaluation Method	Accuracy		
					SPCA-HOG-HOF	Proposed	Difference
	Weizmann (a)	10 / 93	60~70	LOOCV	100%	*100%*	**+0.0%**
	KTH (b)	6 / 2,391	50~70	3-fold	85.6%	*90.9%*	**+5.3%**
	UT-Tower (c)	9 / 108	30~40	LOOCV	96.3%	*98.2%*	**+1.9%**
	VIRAT Aerial (d)	6 / 42	10~40	LOOCV	34.3%	*38.3%*	**+4.0%**

Table 5.3: The comparison between the proposed algorithm in this chapter and the space-time joint feature descriptor (SPCA-HOG-HOF) described in Chapter 4 on the 4 datasets.

The proposed AS is used to describe the temporal evolution of the ASSTID emitted likelihood spectra. The speech-like representation and recognition scheme offer two major advantages. First, we transform an activity sequence into simultaneous temporal signals, which enable us to analyze activities with signal processing techniques (*e.g.* Section 5.1.3). Second, we model activities as the composition of speech, which facilitates the evaluation of higher level activities with linguistic-like models. Our method demonstrates the feasibility of representing human activities as speech-like signals, which enables the further analysis of activities by various state-of-the-art speech recognition technologies.

Chapter 6

Human-Vehicle Interaction Recognition Without Event-Level Training

Recognizing human-vehicle interactions is a challenging problem in computer vision. It is of interest in security, automated surveillance, and aerial video analysis. For example, the detection of a person getting into a vehicle may provide the first level alert of abnormal events. The discovery of frequent human-vehicle interactions from aerial video may help pinpoint a warehouse or signify the migration of a group of people. As shown in Figure 6.1, due to limited image resolution, air turbulence, cloud coverage, objects temporarily out of field of view, and the constantly moving aerial vehicle, the recognition of human-vehicle interactions from aerial view is a much more challenging task than those in normal scenarios. In this work, we propose a general framework to recognize human-vehicle interactions from an aerial video. More specifically, we illustrate our framework using the cases of recognizing a person getting into and out of a vehicle.

With careful and sometimes repeated inspections, a human observer can recognize human-vehicle interactions from aerial video without seeing any examples from the same setup. This is because humans are capable of con-

Figure 6.1: (a) The aerial image of a person approaching the front door of a vehicle. The bounding box of the person is magnified to illustrate this challenging scenario. (b) The snapshots of a vehicle taken from an UAV in every 5 seconds.

stantly tracking objects in low quality imagery and are proficient at reasoning about the underlying event without seeing it in its entirety. However, there are two major difficulties for machine vision to perform the same task as well. First, most machine learning algorithms require a sufficient number of training samples to perform reliable recognition; however, the cost is high for taking aerial videos and annotating example sequences. Second, the key moments of human-vehicle interactions always happen when persons are in close proximity of the vehicle; as a result, a human tracker is easily subject to drift due to overlapped object structures in blurry low-resolution imagery.

Our method is a temporal logic based approach which does not require the tracking of human objects nor event-level training examples. Our system starts with processing the bounding box sequences of the tracked vehicles. To estimate the location and the orientation of a vehicle, we train SVM classifiers

with samples rendered from 3-D vehicle models and ray tracing. Then we search for the optimal solution of vehicle states in a sequence of frames using dynamic programming under a Markovian assumption. Given the aligned 3-D vehicle models, we use the localized door (or trunk) regions together with local human detection results to reason about their interactions over time. We define the temporal flow of a human-vehicle interaction based on the sub-events of particular changes in their spatial relationships. Weights are manually assigned to the interaction associated sub-events according to their relative importance to the composition of the interaction. The likelihood of individual interactions is computed by matching an observation sequence with the formal event representations and binning the weighted votes of matched sub-events. To the best of our knowledge, our work is the first research which explicitly tackles the problem of recognizing human-vehicle interactions in aerial video.

This chapter is arranged as follows. Section 6.1 introduces the technical details of our dynamic programming based 3-D vehicle alignment. Our temporal logic based interaction recognition scheme is presented in Section 6.2. We demonstrate the experimental results in Section 6.3 and conclude in Section 6.4.

6.1 Alignment of 3-D Vehicle Model

The robust alignment of a 3-D vehicle model is essential for the system to extract event ROI and to estimate the human-vehicle spatial relationship. In this section, we propose a novel and generic approach for the optimal search of

vehicles states by the alignment of 3-D vehicle models. In the following subsections, we explain the details of our methodology from (1) 3-D model rendering, (2) localization of a vehicle centroid, (3) estimation of vehicle orientation, and (4) the optimal search of vehicle states using dynamic programming.

6.1.1 3-D Vehicle Model

Collecting training samples for vehicle detection is a tedious task, and it is impractical to collect them in all possible view points. Therefore, we use ray tracing with 3-D vehicle models to generate controlled training images with detailed annotations. In order for our ray tracer to generate synthetic training samples, we create the scene of vehicles using the following descriptions: we place a vehicle model in the center of a 3-D space and a ground plane model below the vehicle model. Then, four point light sources are placed on the front, rear, left, and right of the vehicle model, respectively. Finally, a scene camera is added and controlled by the system as shown in Figure 6.2. By adjusting the position and direction of the camera, our ray tracer can generate the projected images of a 3-D vehicle in different orientations.

Without loss of generality, our ray tracer disables reflection and refraction. It is not possible for the system to simulate the detailed characteristics of the texture of vehicles and the ground from most aerial video data due to low resolution scenes and compression errors.

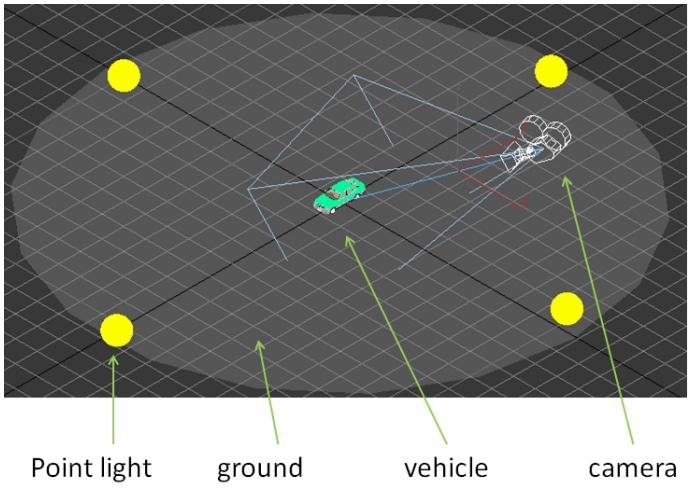


Figure 6.2: A ray tracer with 3-D scene including a vehicle.

6.1.2 Vehicle Location Detection

In this subsection, we explain the probabilistic approach to localize the centroid of the vehicle. Here, we assume that a vehicle is completely visible in the scene. We train a SVM classifier with HOG features extracted from positive and negative vehicle sample images from 3-D vehicle models. The positive samples are vehicle figure-centric images and the negative sample images either contain part of the vehicles or do not present a vehicle. Therefore, the trained binary SVM classifier estimates the probability of the vehicle located at the center of a testing image.

The positive sample set has 720 images from 360 degree orientations and 2 vehicle types. The size of the projected image of a vehicle varies with respect to the camera views. These training samples are uniformly resized with a minimal margin as shown in Figure 6.3. In this process, we measure

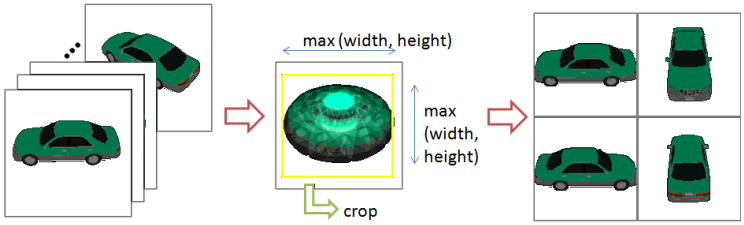


Figure 6.3: Positive vehicle training sample generation.

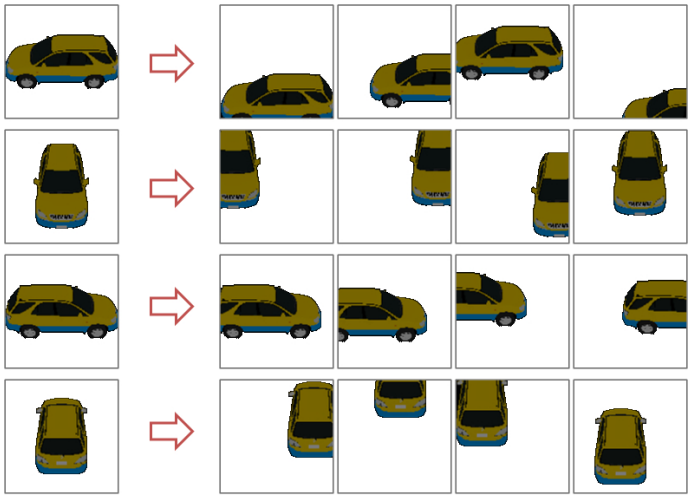


Figure 6.4: Negative vehicle training samples.

the maximum length of the height and width of a vehicle in all orientations, crop the margin, and resize the cropped image. The negative sample set is generated from the positive sample set. For a positive sample set, we generate 4 negative training images from each positive sample by image translation. The displacement vectors are randomly generated in x and y direction. Figure 6.4 illustrates our negative samples.

Extracting descriptive features from the generated training samples is as important as generating robust training samples. The HOG descriptor

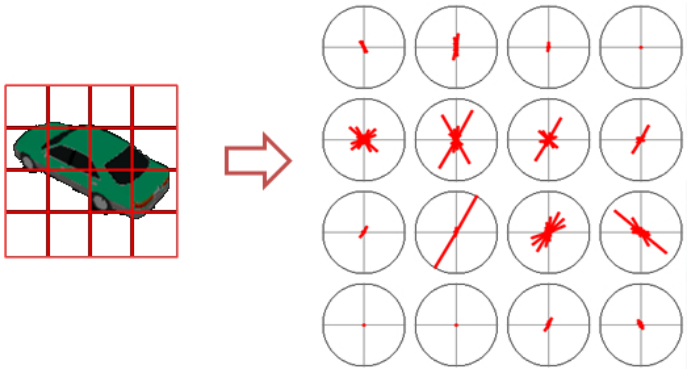


Figure 6.5: The configuration of our HOG descriptor for vehicle location and orientation detection.

has been shown effective in characterizing humans and vehicles. Here, we compute HOG descriptors from image patches using 4×4 cell rectangular cells, 9 orientation bins, and unsigned gradient vectors as shown in Figure 6.5. We train a SVM classifier with HOG descriptors extracted from the positive and negative sample images. The classifier has two classes: (1) *positive*, a vehicle is located in the center of an image and (2) *negative*, a vehicle is not located at the center of an image [73]. In order to compute vehicle figure centric bounding boxes, we scan the input track image by a sliding window to extract the HOG and calculate the probability of a vehicle presenting at the center of the window by the trained SVM classifier.

6.1.3 Vehicle Orientation Estimation

Accurate vehicle orientation estimation enables the extraction of ROI such as door regions after the vehicle location detection. This subsection explains the method to estimate vehicle orientation in the resolution of 10°.

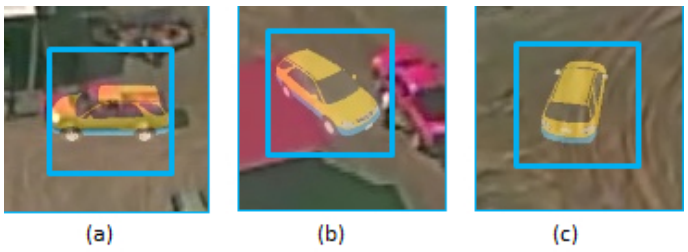


Figure 6.6: Vehicle orientation estimation results.

Our technique of vehicle orientation estimation is similar to the approach of vehicle location detection in that both methods use generated images from a ray tracer with 3-D vehicle models and extract HOG descriptors from synthetic images.

We train a SVM vehicle orientation classifier with 720 images and their HOG descriptors from positive samples of vehicle location detection. The classifier has 36 classes representing every 10° so that each class has 20 training images. The SVM classifier computes the probabilities of vehicle orientations in the testing images. Our SVM classifier performs correctly when the vehicle is located at the center of testing images (Figure 6.6(a)). If a vehicle is not correctly localized (Figure 6.6(b)) or does not exist in the testing images (Figure 6.6(c)), the estimation of our classifier can be erroneous. Therefore, we propose to combine the results of vehicle location detection and vehicle orientation estimation for the accurate estimation of vehicle states.

6.1.4 Dynamic Programming for the Optimal Search

In this subsection, we explain the method for the optimal search of vehicle states (location and orientation) over a sequence of frames using dynamic programming. For the event ROI extraction in Section 6.2, searching both the correct location and orientation of a vehicle is required. We first formulate the joint probability of vehicle location and orientation in a single frame under the assumption that vehicle location and orientation are conditionally independent. Then, we formulate the transition probability of vehicle states in two consecutive frames. With the formulated probability model and our dynamic programming solution, we are able to efficiently search for the optimal vehicle state parameters over the vehicle track sequence.

The joint probability of vehicle location (l) and orientation (o) given an image (I), $P(l, o|I)$, is represented as a product of the probability of vehicle location, $P(l|I)$, and vehicle orientation given vehicle location, $P(o|l, I)$ as shown in Eq. 6.1. The estimation of $P(l|I)$ and $P(o|l, I)$ are explained in Subsection 6.1.2 and 6.1.3.

$$\begin{aligned} P(l, o|I) &= \frac{P(l, o, I)}{P(I)} = \frac{P(l, I)}{P(I)} \cdot \frac{P(l, o, I)}{P(l, I)} \\ &= P(l|I) \cdot P(o|l, I) \end{aligned} \tag{6.1}$$

We formulate the joint probability model of a sequence of the vehicle states given a sequence of track images, $P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}})$, under the Markovian assumption. The variable subscript indicates the frame number. Let $S = \{l, o\}$, which indicates a vehicle state composed of l and o. Then, $P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}})$

can be simplified as $P(S_{\{1,t\}}|I_{\{1,t\}})$. $P(S_{\{1,t\}}|I_{\{1,t\}})$ is expanded by using Bayes' Theorem as shown in Eq. 6.2

$$\begin{aligned} P(S_{\{1,t\}}|I_{\{1,t\}}) &= \frac{P(S_{\{1,t\}}, I_{\{1,t\}})}{P(I_{\{1,t\}})} \\ &= \frac{P(S_t|S_{\{1,t-1\}}, I_{\{1,t\}})P(S_{\{1,t-1\}}, I_{\{1,t\}})}{P(I_{\{1,t\}})} \end{aligned} \tag{6.2}$$

In Eq. 6.2, the term $P(S_{\{1,t-1\}}, I_{\{1,t\}})$ can be expanded as $P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) \cdot P(I_t)$, and the term $P(S_t|S_{\{1,t-1\}}, I_{\{1,t\}})$ can be simplified as $P(S_t|S_{t-1}, I_t)$ by the Markovian assumption. Also, $P(I_t)$ and $P(I_\{1, t\})$ are counted as constants given a sequence of images. Therefore,

$$\begin{aligned} &P(S_{\{1,t\}}|I_{\{1,t\}}) \\ &\quad \propto P(S_t|S_{t-1}, I_t)P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) \end{aligned} \tag{6.3}$$

In Eq. 6.3, the left term can be expanded as the following by using the Bayes' Theorem:

$$\begin{aligned} &P(S_t|S_{t-1}, I_t) \\ &\quad = P(S_t|S_{t-1})P(S_t|I_t)\frac{P(S_t)}{P(I_t)P(S_{t-1})} \end{aligned} \tag{6.4}$$

The right term can also be expanded as the following by using the Bayes' Theorem:

$$\begin{aligned} &P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) \\ &\quad = P(S_{\{1,t-1\}}|I_{\{1,t-1\}})P(I_{\{1,t-1\}}) \end{aligned} \tag{6.5}$$

Under the assumption of the uniform prior probability distribution for S, Eq. 6.3 can be represented as in Eq. 6.6 by Eq. 6.4 and Eq. 6.5.

$$P(S_{\{1,t\}}|I_{\{1,t\}}) \propto P(S_t|S_{t-1})P(S_t|I_t)P(S_{\{1,t-1\}}|I_{\{1,t-1\}}) \tag{6.6}$$

By induction, Eq. 6.6 can be the product of a sequence of terms as shown in Eq. 6.7.

$$P(S_{\{1,t\}}|I_{\{1,t\}}) = P(S_1|I_1)\prod_{k=2}^{k=t}[P(S_k|S_{k-1})P(S_k|I_k)] \tag{6.7}$$

By replacing back S by l and o, we can derive the following equation:

$$P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}}) = P(l_1, o_1|I_1)\prod_{k=2}^{k=t}[P(l_k, o_k|l_{k-1}, o_{k-1})P(l_k, o_k|I_k)] \tag{6.8}$$

$P(l_k, o_k|l_{k-1}, o_{k-1})$ implies the transition probability of vehicle states in two consecutive frames, k and $k-1$. $P(l_k, o_k|I_k)$ is derived from Eq. 6.1. We assume that the transition probability model has the exponential distribution as follows:

$$P(l_k, o_k|l_{k-1}, o_{k-1}) = \lambda_l \cdot \lambda_o \cdot exp(-\lambda_l \cdot \|l_k, l_{k-1}\| - \lambda_o \cdot \|o_k, o_{k-1}\|) \tag{6.9}$$

We model the search for the optimal sequence of vehicle states as a Markov decision process. To limit the search space, the vehicle location is

searched over a uniform grid of 5-pixel resolution, orientation resolution is 10°, and the frame rate is reduced to 2.5 fps from the original 10 fps.

The value iteration V to find the optimal vehicle state sequence is represented as follows:

Initialize $V(S_k)$ *arbitrarily*

loop for frame k

loop for states at k, $S_k = (l_k, o_k)$

loop for states at $k-1$, S_{k-1}

$$V(S_k) = max_{S_{k-1}} \{ S_P(l_1, o_1|I_1) \cdot \prod_{k=2}^{k=t} (P(l_k, o_k|l_{k-1}, o_{k-1}) \cdot P(l_k, o_k|I_k)) \}$$

end loop

end loop

end loop

Via our dynamic programming formulation, the search for the optimal vehicle state sequence improves the state estimation in individual frames. When real-time process is required, our system provides the optimal solution in the current frame. Without the time constraints, the optimal vehicle states in previous frames can be updated using a backward search.

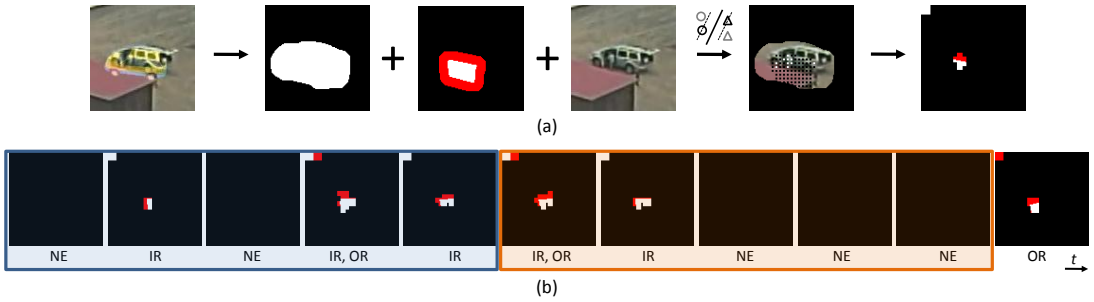


Figure 6.7: (a) The illustration of our human detection process. (b) Our system extracts interaction associated sub-events from a labeled human-vehicle sequence using a two-sided sliding window. The sliding window detects *Meets*(IR,NE), which contributes a weighted vote to the interaction of a person getting into a vehicle.

6.2 Temporal Logic for Human-Vehicle Interaction Recognition

In this section, we introduce our temporal logic based approach, which derives the most likely human-vehicle interaction from low-level information. The low-level processing results include the localized event ROI and the locations of detected human objects, which are assigned with object states and parsed with modified temporal logic for interaction analysis.

6.2.1 Human Detection

After the process of 3-D vehicle model alignment, we perform human detection on the event ROI. As shown in Figure 6.7(a), for the recognition of a person getting into and out of a vehicle, our 3-D vehicle alignment provides the binary masks of the vehicle and its door regions. We dilate both types of masks and apply the vehicle mask to the bounding box so that arbitrary

image content around the vehicle will contribute less to the human detector. The door mask after dilation is marked with a different color to indicate the peripheral of the ROI, which is used to capture a person's approach of ROI.

We use HOG to characterize human objects in low-resolution imagery. Our SVM based human detector is trained with HOG features extracted from manually cropped figure-centric bounding boxes and negative samples from patches around the figures. To save computation, the SVM window classifier only performs detection on grid locations of the event ROI. We train linear SVM to compute calibrated likelihood values [85], which are thresholded to indicate the likely grid locations of human presence. However, the detection accuracy inevitably suffers from the blurry low-resolution imagery as in Figure 6.7(a). Therefore, instead of taking the risk of missing true detections, a low threshold (< 0.5) is used to allow a certain amount of false positives. We perform connected component analysis on the detection grid coordinate to label the detected persons and remove unlikely blobs by area.

To identify the human-vehicle spatial relationship in each bounding box, the dilated mask of event ROI is applied to the mask of human blobs. Based on the overlapped mask, our system estimates whether the person is inside the ROI (IR), outside the ROI (OR), or does not exist (NE) in the image patch. The specific permutations of these three event states are defined as the constituent sub-events of interactions.

6.2.2 Piecewise Temporal Logic

In Allen and Ferguson's classic temporal interval representation of events [6], an event is defined as having occurred if and only if the sequence of observations matches the formal event representation and satisfies the pre-defined temporal constraints. Temporal logic based approaches have been successfully applied for the recognition of human activities, human-human interactions, human-object interactions, and group activities [3]. Most importantly, instead of learning events from training examples, temporal logic allows the direct encoding of human knowledge. However, the recognition of interaction related sub-events from aerial video is far less accurate than that in regular scenarios. Therefore, capturing human-vehicle interactions by matching them against their complete event representation is rarely a success in our experiments.

We adopt a modified temporal logic approach to mine the pieces of event evidence embedded in a human-vehicle sequence. We name our method piecewise temporal logic (PTL), which is different from the classic temporal logic in two major aspects. First, our interaction representation is defined based on event states, from which the higher level interaction associated sub-events are derived. Second, our method recognizes interactions by comparing the weighted sums of detected sub-events, the temporal relationships among which are *not* taken into account. We found that in a human-vehicle sequence, the moments of interaction related primitive actions are not always observable and cannot be reliably recognized. Therefore, we define human-vehicle interactions in terms of the event states that lead to them. Figure 6.8 shows the

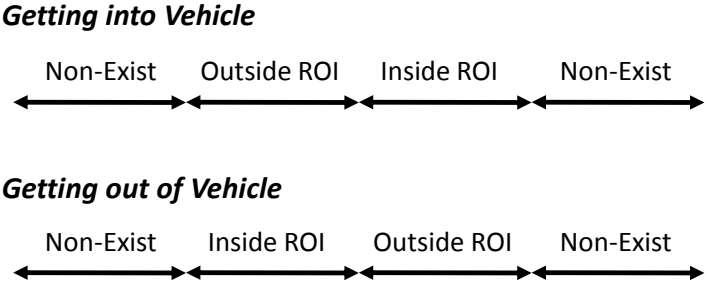


Figure 6.8: The formal event representation of a person getting into and out of vehicle.

formal event representation of a person getting into and out of a vehicle. Given the temporal flows of event states, interaction associated sub-events are defined in terms of the alternations of specific states. The set of predicates we used to describe the temporal relationships of event states include *Meets*, *Starts*, and *Finishes*. These sub-events are manually assigned with weights based on their relative importance to the actual occurrence of the interaction. For example, in Figure 6.7(b), the alternation of event states from IR to NE is more informative than the change from NE to OR for the detection of a person getting into a vehicle. Table 6.1 shows the interaction associated sub-events and their corresponding weights. Note that the exact values of sub-event weights cause much less effect on the system performance than their relative values.

It is a difficult task to extract instances of sub-events from a noisy event state sequence such as Figure 6.7(b). We propose to use a two-sided sliding window to detect interaction associated sub-events. As shown in Figure 6.7(b), the sub-event *Meets*(IR,NE) extracted from rear and front sliding windows is compared with the human encoded list in Table 6.1. The matched sub-event

Interaction	Sub-event	Weight
Getting into vehicle	*Meets*(IR,NE)	2
	Meets(OR,IR)	1
	Meets(OR,NE)	0.5
	Finishes(IR)	-2
Getting out of vehicle	*Meets*(NE,IR)	2
	Meets(IR,OR)	1
	Meets(NE,OR)	0.5
	Starts(IR)	-2

Table 6.1: Interaction associated sub-events and their corresponding weights. IR, OR, and NE are shorts for human inside the ROI, outside the ROI, and does not exist (NE) in the image bounding box, respectively. *Meets*, *Starts*, and *Finishes* are the temporal predicates used to define their relationships.

contributes a weighted vote to the corresponding bin of an event histogram. We use the sum of absolute sub-event weights in an event histogram to determine if any of the two interactions have ever occurred. The normalized event histogram indicates the occurrence likelihood of interactions.

6.3 Experimental Results

We test our methodology with the challenging VIRAT Aerial Video dataset [58]. The videos were taken in 30 frames per second with the resolution of 720 by 480 pixels. As shown in Figure 6.9, the challenges posed by this dataset include low image resolution, vague object appearance and motion (due to air turbulence and video compression artifacts), time-varying views, changing weather conditions, salient shadow, and cluttered backgrounds.

There are a number of human-vehicle sequences in this dataset. How-

Figure 6.9: The snapshots of four true positive (TP), two true negative (TN), one false negative (FN), and one false positive (FP) sequence are shown. We treat the subject human-vehicle interactions (getting into vehicle, getting out of vehicle) as the positive class and all other events (others) as the negative class.

ever, we can only find 7 instances of a person getting into and out of a vehicle. We manually select 20 other types of human-vehicle interaction sequences, in which a person may be passing by or (un)loading the vehicle. Therefore, in our evaluation set, there are 4 sequences of a person getting into a vehicle, 3 sequences of a person getting out of a vehicle, and 20 other types of human-vehicle sequences. We use the same set of parameters for vehicle alignment and interaction analysis without any event-level training. Figure 6.9 shows the snapshots of our testing sequences. Despite the differences in the types of vehicles, viewpoints, and interactions, our system is able to correctly detect the subject human-vehicle interactions from sequences such as the TP examples in Figure 6.9. The FP and FN examples in Figure 6.9 show the cases when our method fails. In the sequence of *"Getting into vehicle, FN"*, the approach of the person from the left was partially occluded by the building, and in the sequence of *"Others, FP"* the departure of the person from the ROI misled the system.

Our system demonstrates superior results on the search of the optimal vehicle states. In 20 sequences out of 27 testing sequences (74.1%), both the orientation and location of vehicles are correctly estimated. In the 6 instances out of 7 incorrect sequences (22.2%), the locations of the vehicles are correctly detected but the vehicle orientations are 180° reversed. In spite of that, the ROI in those sequences were correctly located because of the symmetry of vehicle shape. In the other 1 instance (3.7%), the estimation of the vehicle orientations is incorrect. For interaction recognition, we analyze sub-

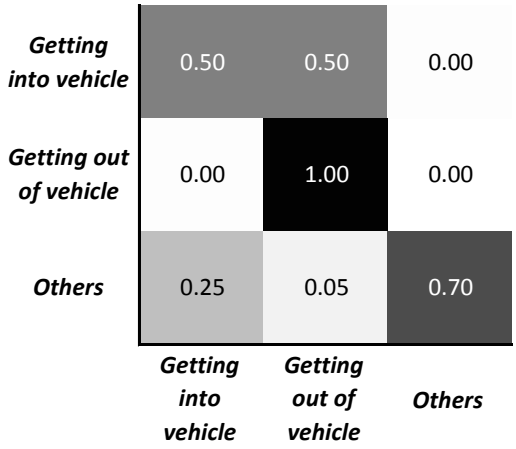


Table 6.2: The confusion matrix of our method on a subset of the VIRAT Aerial Video dataset.

events in every 4-second long two-sided sliding window. The system classifies a sequence as the subject human-vehicle interactions if its sum of absolute sub-event weights exceeds 1 and there is no tie in the event histogram. A sequence is recognized as other events if the sum of absolute sub-event weights is less than 1 or there is a tie in its event histogram. Table 6.2 shows the confusion matrix. By treating the subject human-vehicle interactions as the positive class and all other events as the negative class, the accuracy of our method on this evaluation set is 77.78% ((TP + TN) / (TP + TN + FP + FN)), the precision is 53.85% (TP / (TP + FP)), and the recall is 100.0% (TP / (TP + FN)).

6.4 Conclusions

We propose a general framework for the recognition of human-vehicle interactions from aerial view. Our method offers three major advantages to better resolve the challenges posed in this scenario. First, we adopt a temporal logic based approach to avoid the cost of manually collecting and labeling the training examples. Second, we employ a dynamic programming based 3-D vehicle model alignment technique, which accurately locates event ROI with the consideration of the previous alignment results. Third, based on classic temporal logic, we introduce the concept of PTL, which significantly improves the recognition performance in our problem. PTL detects interaction sub-events by checking the temporal relationships between the event states. However, at the semantic-level, the temporal logics among the sub-events are not verified to induce the robustness against sequences of noisy sub-events. Furthermore, the proposed method can be generalized to recognize any kinds of human-vehicle interactions with the proper encoding and weighting of the temporal logics between event states. Most importantly, our method demonstrates high recognition accuracy on the challenging VIRAT Aerial Video dataset.

Chapter 7

Conclusions

We have presented our approaches for track sequence preprocessing, human activity recognition, and human-vehicle interaction recognition with the challenges from low-resolution imagery taken into account. The major challenges include inaccurate tracking of objects, time varying human cast shadows, and sparse and coarse low-level features. Existing methods that do not specifically address these issues usually show near random performance on real-world low-resolution video datasets (*e.g.* VIRAT Aerial Video dataset [58]).

To refine human tracking results, we train a SVM classifier with figure-centric human bounding boxes to localize the image patch which is most likely to be centered on the tracked person. We also propose a shadow removal technique which recovers the color of the background region shaded by a human figure. To reduce the chances of removing pixels from human figures, our method adopts a bottom-up approach to detect a human cast shadow as a connected region instead of scattered pixels.

The choice of low-level feature representations predominates the performance of an activity recognition system. This statement is especially valid

in our problem. The accurate computation of low-level features is not feasible in this scenario; therefore, we employ histogram based descriptors to make use of noisy gradient and optical flow measurements. Despite the robustness of histogram based representation of pixel-level shape and motion features for activity characterization, there is more to be exploited in terms of type of information. For example, we commonly see specific body part movements or gestures occurring across the video sequences of the same action. Based on that observation, we propose action spectrogram to represent the spectral properties of action associated local visual patterns computed from active body parts. Our novel mid-level feature captures the occurrence of action specific visual patterns over time, and enables us to model an activity sequence as concurrent speech-like signals.

For the inference of high-level human vehicle interactions from low-resolution videos, we devise our algorithm in a way that reduce the complexity and hierarchy of the processes. At the low-level, we train separate detectors to estimate vehicle location and orientation instead of relying on foreground masks or edge maps for template matching. We integrate the detectors' response into a dynamic programming formulation so that the optimal alignment of a 3-D vehicle is computed over an image sequence. The use of synthetic 3-D vehicle models provides us the location estimate of an event ROI, which restricts the image area to be searched for the possible existence of human objects. In addition, we localize human objects by per frame detection rather than maintaining tracks, which reduces the risks of tracking

humans from cluttered vehicle structures and broken tracks from objects out of FOV. We propose the algorithm of piecewise temporal logic for high-level human vehicle interaction recognition. Different from the previous temporal logic based approaches for human-object interaction recognition, our approach bypasses the direct detection of interaction sub-events, which is highly unreliable when a human is in close contact with a vehicle in low-resolution videos. Instead, the sub-events are defined in terms of the particular alternations of even states based on the detected human positions w.r.t. the localized ROI. We smooth the sequence of event states with a two-sided sliding window and classify human-vehicle interactions with a weighted event histogram. Without the use of fancy event representations or machine learning algorithms, we achieve rather promising results by replacing more feature dependent techniques (i.e. tracking, primitive action recognition) with the inference from less feature dependent processes.

Bibliography

[1] http://en.wikipedia.org/wiki/Activity_recognition.

[2] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. *In CVIU*, 73:428–440, 1999.

[3] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. In *ACM Computing Surveys (CSUR)*, volume 43(3), 2011.

[4] M. A. R. Ahad, J. K. Tan, H. S. Kim, and S. Ishikawa. A simple approach for low-resolution activity recognition. *Intl. J. for Computational Vision and Biomechanics (IJCVB)*, 3(1), 2010.

[5] S. Ali, A. Basharat, and M. Shah. Chaotic invariants for human action recognition. *In ICCV*, 2007.

[6] J. F. Allen and G. Ferguson. Actions and events in interval temporal logic. In *Journal of Logic and Computation*, volume 4, 1994.

[7] S. Biand, D. Liang, X. Shen, and Q. Wang. Human cast shadow elimination method based on orientation information measures. *IEEE International Conference on Automation and Logistics (ICAL)*, 2007.

[8] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *In ICCV*, 2005.

[9] A. Bobick and J. Davis. The representation and recognition of action using temporal templates. *In PAMI*, 23(3), 2001.

[10] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. *In CIVR*, 2007.

[11] B. Chakraborty, M. Pedersoli, and J. Gonzalez. View-invariant human action detection using component-wise hmm of body parts. *Lecture Notes in Computer Science*, 5098, 2008.

[12] C. Chang, W. Hu, J. Hsieh, and Y. Chen. Shadow elimination for effective moving object detection with gaussian models. *In ICPR*, 2002.

[13] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[14] R. Chaudhry, A. Ravichandran, G.D. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. *In CVPR*, 2009.

[15] C.-C. Chen and J. K. Aggarwal. Recognizing human action from a far field of view. In *IEEE Workshop on Motion and Video Computing (WMVC)*, Utah, USA, 2009.

[16] C.-C. Chen and J. K. Aggarwal. Human shadow removal with unknown light source. In *International Conference on Pattern Recognition (ICPR)*, Istanbul, Turkey, 2010.

[17] C.-C. Chen and J. K. Aggarwal. Modeling human activities as speech. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, USA, 2011.

[18] H.T. Chen, H.-H. Lin, and T.-L. Liu. Multi-object tracking using dynamical graph matching. *In CVPR*, 2001.

[19] Y.-W. Chen and C.-J. Lin. *Combining SVMs with various feature selection strategies*. Springer, 2006.

[20] C. C. Chibelushi, F. Deravi, and J. S. D. Mason. A review of speech-based bimodal recognition. In *IEEE Trans. Multimedia*, volume 4, pages 23 –37, 2002.

[21] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti. Improving shadow suppression in moving object detection with hsv color information. In *IEEE Proceedings of Intelligent Transportation Systems*, pages 334 –339, 2001.

[22] R. Cutler and L. Davis. Robust real-time periodic motion detection, analysis, and applications. In *PAMI*, volume 22, pages 781–796, 1999.

[23] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[24] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. *In ECCV*, 2006.

[25] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *In European Conference on Computer Vision*, 2006.

[26] K. H. Davis, R. Biddulph, and S. Balashek. Automatic recognition of spoken digits. In *J. Acoust. Soc. Am.*, 1952.

[27] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. *In ICCV-W VS-PETS*, 2005.

[28] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. *In ICCV*, 2003.

[29] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. *In CVPR*, 2008.

[30] G. D. Finlayson, M. Drew, and C. Lu. Intrinsic images by entropy minimization. *In ECCV*, 2004.

[31] H. Fujiyoshi. Real-time human motion analysis by image skeletonization. *IEICE Trans. Info. and Syst.*, 2004.

[32] A. Ganapathiraju, J. E. Hamaker, and J. Picone. Applications of support vector machines to speech recognition. In *IEEE Trans. Sig. Proc.*, volume 52, pages 2348 – 2355, 2004.

[33] D. M. Gavrila. The visual analysis of human movement: A survey. *In CVIU*, 73:82–98, 1999.

[34] J. D. Gibbons. *Nonparametric Methods for Quantitative Analysis*. American Sciences Press, 1985.

[35] K. Guo, P. Ishwar, and J. Konrad. Action change detection in video by covariance matching of silhouette tunnels. In *In ICASSP*, 2010.

[36] R. Guo, Q. Dai, and D. Hoiem. Single-image shadow detection and removal using paired regions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[37] C. Harris and M.J. Stephens. A combined corner and edge detector. *In Alvey Vision Conference*, 2008.

[38] K. Hatun and P. Duygulu. Pose sentences: A new representation for action recognition using sequence of pose words. *In ICPR*, 2008.

[39] B. Herbert, A. Ess, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110:346–359, 2008.

[40] P.-C. Hsiao, C.-S. Chen, and L.-W. Chang. Human action recognition using temporal-state shape contexts. *In ICPR*, 2008.

[41] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *Trans. on Systems, Man, and Cybernetics*, 34:334–352, 2004.

[42] N. Ikizler, R. G. Cinbis, and P. Duygulu. Human action recognition with line and flow histograms. *In ICPR*, 2008.

[43] Y. Ivanov, C. Stauffer, A. Bobick, and W. E. L. Grimson. Video surveillance of interactions. *IEEE Workshop on Visual Surveillance*, 1999.

[44] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. *In ICCV*, 2007.

[45] S. W. Joo and R. Chellappa. Attribute grammar-based event recognition and anomaly detection. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR-W)*, 2006.

[46] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. *In ICCV*, 2005.

[47] I. Laptev and T. Lindeberg. Space-time interest points. *In ICCV*, 2003.

[48] I. Laptev and P. Pérez. Retrieving actions in movies. *In ICCV*, 2007.

[49] J. T. Lee*, C.-C. Chen*, and J. K. Aggarwal. Recognizing human-vehicle interactions from aerial video without training. In *Workshop of Aerial Video Processing in conjunction with CVPR (WAVP)*, Colorado Springs, USA, 2011. (*These two authors contributed equally to the paper.).

[50] J. T. Lee, M. S. Ryoo, and J. K. Aggarwal. View independent recognition of human-vehicle interactions using 3-d models. In *IEEE Workshop on Motion and Video Computing (WMVC)*, 2009.

[51] X. Li. Hmm based action recognition using oriented histograms of optical flow field. *Electronics Letters*, 2007.

[52] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". *In CVPR*, 2009.

[53] D. G. Lowe. Distinctive image features from scale-invariant keypoint. *International Journal of Computer Vision*, 60:91–110, 2004.

[54] W. Lu and J. Little. Simultaneous tracking and action recognition using the pca-hog descriptor. *Canadian Conference on Computer and Robot Vision (CRV)*, 2006.

[55] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(5):530 –549, 2004.

[56] P. Matikainen, M. Hebert, and R. Sukthankar. Trajectons: Action recognition through the motion analysis of tracked features. *In ICCV Workshops*, 2009.

[57] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *IJCV*, volume 79, pages 299–318, 2008.

[58] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. K. Aggarwal, H. Lee, L. Davis, E. Swears, X. Wang, Q. Ji, K. Reddy,

M. Shah, C. Vondrick, H. Pirsiavash, D. Ramanan, J. Yuen, A. Torralba, B. Song, A. Roy-Chowdhury, and M. Desai. A large-scale benchmark dataset for event recognition in surveillance video. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[59] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. In *PAMI*, volume 22, pages 831–843, 2000.

[60] A. Prati, I. Mikic, R. Cucchiara, and M. M. Trivedi. Comparative evaluation of moving shadow detection algorithms. In *In CVPR-EEMCV*, 2001.

[61] V. Reilly, B. Solmaz, and M. Shah. Geometric constraints for human detection in aerial imagery. In *Proceedings of the 11th European conference on Computer vision (ECCV)*, Berlin, Heidelberg, 2010.

[62] M. S. Ryoo and J. K. Aggarwal. Recognition of composite human activities through context-free grammar based representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, New York, USA, 2006.

[63] M. S. Ryoo and J. K. Aggarwal. Semantic representation and recognition of continued and recursive human activities. *In IJCV*, 2009.

[64] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities.

In ICCV, 2009.

[65] M. S. Ryoo, C.-C. Chen, J. K. Aggarwal, and A. Roy-Chowdhury. An overview of contest on semantic description of human activities 2010. *In ICPR Contests*, 2010.

[66] M. S. Ryoo, J. T. Lee, and J. K. Aggarwal. Video scene analysis of interactions between humans and vehicles using event context. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '10, pages 462–469, 2010.

[67] E. Salvador, A. Cavallaro, and T. Ebrahimi. Shadow identification and classification using invariant color models. *In ICASSP*, 2001.

[68] R. Santiago-Mozos, J.M. Leiva-Murillo, F. Perez-Cruz, and A. Artes-Rodriguez. Supervised-pca and svm classifiers for object detection in infrared images. *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2003.

[69] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. *In ICPR*, 2004.

[70] E. Schwartz. Spatial mapping in primate sensory projection: analytic structure and relevance to perception. *Biological Cybernetics*, 25:181–194, 1977.

[71] X. Song and R. Nevatia. A model-based vehicle segmentation method for tracking. In *ICCV*, 2005.

[72] I. Steinwart. Sparseness of support vector machines - some asymptotically sharp bounds. *In NIPS*, pages 169–184, 2004.

[73] B. Tamersoy and J. K. Aggarwal. Robust vehicle detection for tracking in highway surveillance videos using unsupervised learning. In *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2009.

[74] C. Thurau. Behavior histograms for action recognition and human detection. *Lecture Notes in Computer Science*, 4814:299–312, 2007.

[75] S. D. Tran and L. S. Davis. Event modeling and recognition using markov logic networks. In *Proceedings of the 10th European Conference on Computer Vision*, 2008.

[76] V. Tsai. A comparative study on shadow compensation of color aerial images in invariant color models. *IEEE Trans. on Geoscience and Remote Sensing*, 2006.

[77] R. Vezzani, M. Piccardi, and R. Cucchiara. An efficient bayesian framework for on-line action recognition. In *In ICIP*, 2009.

[78] P. Viola and M. Jones. Robust real-time face detection. In *IJCV*, volume 57, pages 137–154, 2004.

[79] C. Wallraven. Recognition with local features: the kernel recipe. *In ICCV*, 2003.

[80] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[81] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. *In BMVC*, 2009.

[82] D. Weinland, R. Ronfard, and E. Boyer. Free viewpoint action recognition using motion history volumes. In *Computer Vision and Image Understanding (CVIU)*, 2006.

[83] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. *In ECCV*, 2008.

[84] M. Woelfel and J. McDonough. *Distant Speech Recognition*. Wiley, 2009.

[85] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. In *Journal of Machine Learning Research*, volume 5, pages 975–1005, 2004.

[86] A. Yao, J. Gall, and L. Van Gool. A hough transform-based voting framework for action recognition. In *In CVPR*, 2010.

[87] E. Yu and J. K. Aggarwal. Human action recognition with extremities as semantic posture representation. *International Workshop on Semantic Learning Applications in Multimedia in association with CVPR*, 2009.

[88] Elden Yu and J. K. Aggarwal. Human extremity detection and its applications in action detection and recognition. In *Pattern Recognition, Machine Intelligence and Biometrics*. Springer and Higher Education Press, 2011.

[89] J. Zhang, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. In *IJCV*, volume 73, 2007.

[90] J. Zhu, K. G. G. Samuel, S. Z. Masood, and M. F. Tappen. Learning to recognize shadows in monochromatic natural images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.